

E.T.S. de Ingeniería Industrial,
Informática y de Telecomunicación

Adecuación y Puesta en Producción de la Descarga Automática de Imágenes Sentinel en el Entorno Tecnológico de Gobierno de Navarra



Grado en Ingeniería
en Tecnologías de Telecomunicación

Trabajo Fin de Grado

Autora: Ainhoa Rojo Bonastre

Director: Eduardo Magaña Lizarrondo

Pamplona, 28 de Junio de 2017

RESUMEN

El presente Trabajo Fin de Grado consiste en el desarrollo de un sistema de descarga automática de imágenes y datos obtenidos por teledetección por parte de las misiones satélites Sentinel, pertenecientes al programa europeo, de observación de la Tierra, Copernicus.

Este sistema deberá adaptarse al entorno tecnológico de Gobierno de Navarra, encargado de aportar una infraestructura eficiente para el proceso de descarga y almacenamiento de imágenes, destinada en el Centro de Procesamiento de Datos.

Con esta finalidad, se ha diseñado una plataforma de descarga multiservidora de código abierto que consta de la suficiente autonomía de tomar decisiones ante la descarga de productos corruptos y fallos en las conexiones con los distintos servidores web, teniendo registros de estos errores. Se aporta además de un interfaz web que facilite la configuración y monitorización de este sistema.

La aplicación desarrollada cuenta con la flexibilidad para no estar limitada a una arquitectura tecnológica concreta ni al uso restringido de Gobierno de Navarra.

ABSTRACT

The current End-of-degree work is focused on the development of an automatic download system of images and data remotely sensed by Sentinel satellite missions, which belong to the Earth Observation European programme Copernicus.

This system should be suited to the technological environment of Navarra's Government. This entity is the one in charge of providing an effective infrastructure for the download and storage process of images, which will be set in the Data Processing Centre.

With this purpose, an open source multiserver download platform has been designed with enough autonomy in order to take decisions in light of the download of corrupted images and web servers connections failures, having records of this errors. Furthermore, a web interface is provided in order to facilitate the configuration and monitoring of the system.

The developed applications counts on the flexibility to not be limited to a specific technological architecture and to the restricted use of Navarra's Government either.



LISTA DE PALABRAS CLAVE

- **Teledetección espacial** – Técnica para la adquisición de imágenes terrestres a distancia
- **Copernicus** – Programa europeo de observación de la Tierra
- **ESA** – Agencia Espacial Europea
- **Sentinel** – Denominación de los satélites del programa Copernicus
- **PyrenEOS** – Proyecto transfronterizo para la gestión de recursos y riesgos naturales
- **CGS** – Collaborative Ground Segment
- **NRT** – Near Real Time
- **OLQC** – Online Quality Control
- **CPD** – Centro de Procesamiento de Datos
- **PostgreSQL** – Sistema de gestión de datos objeto-relacional de código abierto
- **PostGIS** – Extensión de PostgreSQL que añade analítica y manejo de datos espaciales
- **Batch** - Archivo de procesamiento por lotes que contiene comando DOS (Shell de Windows)
- **Python** – Lenguaje de programación de alto nivel, multipropósito e interpretado
- **Psycopg2** – Librería de Python que permite la conexión a PostgreSQL
- **cURL** – Client URL, software con intérprete de línea de comandos orientado a la transferencia multiprotocolo de ficheros
- **Html5** – HyperText Markup Language, lenguaje de marcado para el desarrollo de páginas web y ejecutado en el lado cliente
- **PHP** - HyperText Preprocessor, lenguaje de código abierto para el desarrollo de páginas web y ejecutado en el lado servidor



ÍNDICE DE CONTENIDOS

1. INTRODUCCIÓN.....	10
2. OBJETIVOS.....	11
3. ESTADO DEL ARTE.....	12
3.1. FUNDAMENTOS DE LA TELEDETECCION ESPACIAL.....	12
3.1.1. Teledetección espacial y sus aplicaciones	12
3.1.2. Tipos de resoluciones de un sistema	13
3.1.3. Obtención de una imagen satélite.....	14
3.1.4. Respuesta espectral de la superficie	16
3.2. PROGRAMA COPERNICUS.....	17
3.2.1. Copernicus en breve.....	17
3.2.2. Misiones satélite: familia Sentinel	18
3.2.2.1. Sentinel 1	19
3.2.2.2. Sentinel 2	19
3.2.2.3. Sentinel 3.....	19
3.2.2.4. Sentinel 4.....	20
3.2.2.5. Sentinel 5	20
3.2.2.6. Sentinel 5 Precursor	20
3.2.2.7. Sentinel 6.....	20
3.2.3. Productos Sentinel.....	20
3.3. PROYECTO PYRENEOS	22
3.3.1. Visión general	22
3.3.2. Servicios	22
3.3.3. Subsistemas de la plataforma PyrenEOS	23
4. SOFTWARE DE DESCARGA AUTOMÁTICA DE IMÁGENES SENTINEL.....	25
4.1. PLATAFORMA PYRENEOS, ENTORNO TECNOLÓGICO E INFRAESTRUCTURA DE GOBIERNO DE NAVARRA.....	25
4.1.1. Requisitos tecnológicos.....	25
4.1.2. Plataforma software diseñada y Escenario hardware planteado.....	26
4.1.3. Escenario simulado.....	30
4.2. ESTUDIO COMPARATIVO DE REPOSITORIOS WEB DE PROVEEDORES	31
4.2.1. Necesidad.....	31
4.2.2. Puntos de Acceso Web y Tipología	32
4.2.3. Requisitos a cumplir y primeros descartes.....	33
4.2.4. Herramientas y Metodología utilizadas.....	36
4.2.5. Entorno de evaluación.....	39
4.2.6. Comparativas	40
4.2.6.1. Productos Sentinel disponibles	40



4.2.6.2. Tiempos de adquisición	41
4.2.6.3. Tiempos de descarga	44
4.2.7. Conclusiones de repositorios web	53
4.3. DISEÑO Y DIAGRAMA DE FLUJO DEL PROGRAMA	55
4.4. DESARROLLO DEL PROGRAMA	59
4.4.1. Interfaz web de Configuración y Monitorización	59
4.4.2. Descarga de Parámetros y Automatización	62
4.4.3. Descargador de Imágenes Multiservidor	66
4.4.4. Comprobador de Imágenes Corruptas	67
4.4.5. Carga a Base de Datos PostGIS y Almacenamiento	69
4.4.6. Seguimiento de Estado y Errores	71
4.5. FUNCIONAMIENTO Y RESULTADOS	72
4.5.1. Puesta en marcha de la aplicación.....	72
4.5.2. Resultados	72
5. CONCLUSIONES	74
BIBLIOGRAFÍA	76
ANEXO A: PROGRAMACIÓN DE TAREAS DE WINDOWS	79
ANEXO B: MANUAL DE INSTALACIÓN EN EL SERVIDOR DE DESCARGA	82
ANEXO C: MANUAL DE INSTALACIÓN EN EL SERVIDOR WEB.....	84
ANEXO D: MANUAL DE USO DEL PORTAL WEB	96



ÍNDICE DE FIGURAS

Figura 1 : Diagrama de los elementos del proceso de percepción remota o teledetección. <i>Fuente: INEG [2]</i>	12
Figura 2: Características del espectro electromagnético. <i>Fuente: Wikipedia [3]</i>	14
Figura 3: Concepto de bandas. <i>Fuente: INEG</i>	14
Figura 4: Representación teledetección pasiva y activa. <i>Fuente: GrindGIS [6]</i>	15
Figura 5: Composición de una imagen satélite y de un píxel con varios objetos. <i>Fuente: Teledet [4]</i>	15
Figura 6: Comparación de la signatura espectral del agua, suelo y vegetación. <i>Fuente: SEOS (Science Education through Earth Observation for High Schools) [7]</i>	16
Figura 7: Comparativa de la reflectancia entre una planta sana y una estresada. <i>Fuente: NOAA (National Oceanic and Atmospheric Administration) [8]</i>	16
Figura 8: Visión general del sistema del programa Copernicus. <i>Fuente: Copernicus [9]</i>	17
Figura 9: Puntos de acceso a los servicios Copernicus y a las infraestructuras de datos satélite. <i>Fuente: Copernicus [9]</i>	18
Figura 10: Familia de satélites Sentinel. <i>Fuente: ESA [11]</i>	18
Figura 11: Programa de lanzamiento de satélites Sentinel. <i>Fuente: Copernicus [9]</i>	19
Figura 12: Esquema de niveles de productos por modos del Sentinel 1. <i>Fuente: ESA [14]</i>	20
Figura 13: Esquema y características de niveles de productos del Sentinel 2. <i>Fuente: ESA [15]</i>	21
Figura 14: Esquema y características de niveles de productos por modos del Sentinel 3. <i>Fuente: ESA [16]</i>	21
Figura 15: Arquitectura software propuesta para la plataforma navarra PyrenEOS. <i>Fuente: Álvaro Huarte. Tracasa [19]</i>	26
Figura 16: Escenario diseñado para la descarga y almacenamiento del proyecto PyrenEOS.....	29
Figura 17: Recreación del escenario servidor-almacenamiento a simular	30
Figura 18: Ejemplos servicios de mantenimiento en SCIHUB. <i>Fuente: Copernicus Open Access Hub [24]</i>	31
Figura 19: Imagen no válida del satélite Sentinel 2A con pérdida de datos.	32
Figura 20: Estructura de contenidos de productos Sentinel 2. <i>Fuente: USGS [26]</i>	34
Figura 21: Estructura de contenidos de productos Sentinel 2 en AWS.....	36
Figura 22: Estructura general de los archivos existentes por cada repositorio dentro de Web_Repositories_Test	37
Figura 23: Código ejemplo en Python para la extracción del tiempo de descarga y la fecha de obtención de un nuevo producto	38
Figura 24: Código en Python para cambiar de forma dinámica la fecha de inicio de período de captura de satélite	38
Figura 25: Diagrama de funcionamiento de las herramientas para los estudios de tiempos en cada servidor.....	39
Figura 26: Diagrama de priorización de los repositorios para los distintos tipos de productos	54
Figura 27: Ejemplos de las indicaciones y datos a cambiar por el usuario en los distintos .bat de DOWNLOAD_SERVER.....	56
Figura 28: Arquitectura y flujo cliente-servidor del sistema de descarga	57



Figura 29: Diagrama de flujo general entre servidor web, base de datos y servidor de descarga	58
Figura 30: Diagrama de decisión de los php dedicados a las configuraciones del sistema de descarga	61
Figura 31: Declaración SQL para la creación de una tabla a través de php	61
Figura 32: Ejemplo del uso correcto de la librería psycopg2 para la conexión a la base de datos.....	62
Figura 33: Diagrama de flujo del bloque de Descarga de Parámetros y Automatización, script check_db.py	63
Figura 34: Funciones para la automatización de estudios en check_tools.py.....	64
Figura 35: Funciones para la automatización de descarga en check_download.py.....	65
Figura 36: Diagrama de flujo del proceso de descarga desde un repositorio.....	66
Figura 37: Estado del control de calidad en el archivo MTD_MSIL1C.xml	68
Figura 38: Diagrama de flujo de la descompresión y comprobación de errores de los productos	69
Figura 39: Diagrama de flujo de almacenamiento en disco y base datos de los productos	70
Figura 40: Ejemplo del contenido del fichero autoprogramable para la descarga de productos Sentinel 2	73
Figura 41: Tarea programada de la descarga desde check_db.py.....	73
Figura 42: Ejemplo del almacenamiento de un producto tras su correcta descarga	73



ÍNDICE DE TABLAS

Tabla 1: Herramientas de los servicios VERDE de la plataforma PyrenEOS.	22
Tabla 2: Herramientas de los servicios AZUL de la plataforma PyrenEOS.	23
Tabla 3: Cantidad media de imágenes y capacidad requerida anual de Sentinel 1A y 1B	27
Tabla 4: Cantidad media de imágenes y capacidad requerida anual de Sentinel 2A	27
Tabla 5: Cantidad media de imágenes y capacidad requerida anual de Sentinel 3A-LAND	28
Tabla 6: Cantidad media de imágenes y capacidad requerida anual de Sentinel 3A-OCEAN	28
Tabla 7: Listado y Clasificación de los Punto de Acceso a imágenes Sentinel existentes	32
Tabla 8: Tipos de imágenes disponibles según la misión satélite en los distintos servidores	40
Tabla 9: Tipos de imágenes disponibles según el nivel de producto en los servidores a estudiar	41
Tabla 10: Límite estimado en cada servidor para el proceso de descarga de PyrenEOS	55
Tabla 11: Privilegios según el tipo de usuario en Sentinel Product Ingestor	60
Tabla 12: Tipos y asuntos de registros notificados por el programa de descarga	71



ÍNDICE DE GRÁFICAS

Gráfica 1: Eje cronológico de adquisición de productos SENTINEL 1 de los servidores web.....	42
Gráfica 2: Eje cronológico de adquisición de productos SENTINEL 2 de los servidores web.....	42
Gráfica 3: Eje cronológico de adquisición de productos SENTINEL 3 de los servidores web.....	43
Gráfica 4: ASF- Media estadística de los tiempos de descarga a lo largo del día	44
Gráfica 5: ASF- Velocidades de descarga a lo largo del día	45
Gráfica 6: AUSTRIANM- Media estadística de los tiempos de descarga a lo largo del día	45
Gráfica 7: AUSTRIANM- Velocidades de descarga a lo largo del día	45
Gráfica 8: CODA- Media estadística de los tiempos de descarga a lo largo del día	46
Gráfica 9: CODA- Velocidades de descarga a lo largo del día	47
Gráfica 10: CODE-DE- Media estadística de los tiempos de descarga a lo largo del día	47
Gráfica 11: CODE-DE- Velocidades de descarga a lo largo del día.....	48
Gráfica 12: HNSDMS- Media estadística de los tiempos de descarga a lo largo del día	48
Gráfica 13: HNSDMS- Velocidades de descarga a lo largo del día.....	49
Gráfica 14: PEPS – Media estadística de los tiempos de descarga a lo largo del día.....	49
Gráfica 15: PEPS- Velocidades de descarga a lo largo del día.....	50
Gráfica 16: SCIHUB- Media estadística de los tiempos de descarga a lo largo del día.....	50
Gráfica 17: SCIHUB – Velocidades de descarga a lo largo del día.....	51
Gráfica 18: THEIA- Media estadística de los tiempos de descarga a lo largo del día	51
Gráfica 19: THEIA- Velocidades de descarga a lo largo del día.....	52
Gráfica 20: Comparativa de velocidades medias de descarga de los distintos servidores.....	52
Gráfica 21: Comparativa de velocidades mínimas de descarga de los distintos servidores	53



1. INTRODUCCIÓN

El gran avance tecnológico que se ha dado a lo largo de los últimos años, destacando el crecimiento en el sector de las TIC, ha propiciado en Europa la creación de numerosos programas de financiación orientados a la innovación e investigación aplicada.

Gracias a este impulso se ha conseguido dar un paso más adelante en la observación de la Tierra con la creación del programa Copernicus en 2014. Pero, ¿qué tiene de especial? En el espacio actualmente hay decenas de satélites que son capaces de vigilar la Tierra pero ninguno envía datos siempre, de forma continua, pasando un mismo satélite de Copernicus cada pocos días por una misma zona.

Además, antes de este, disponer de una imagen satélite con unos mínimos de calidad suponía un elevado coste lo que significaba que muchos usuarios contaban con un escaso número de ellas, haciendo más complicado el estudio de su evolución temporal. Copernicus supone una revolución tecnológica que permite obtener de forma gratuita un conjunto de imágenes de alta calidad. Esto conseguirá que numerosas empresas, especialmente pymes y startups tecnológicas, puedan utilizar estos datos para desarrollar sus negocios.

Una de las iniciativas europeas es el Programa de Cooperación Territorial INTERREG, concretamente el programa operativo INTERREG POCTEFA [0] creado para fomentar el desarrollo fronterizo entre España (provincias del norte pirenaico), Francia (provincias del Pirineo francés) y Andorra. Sus prioridades de inversión se centran en reforzar la cooperación en materia I+D+i y favorecer el desarrollo de tecnologías innovadoras en materia de recursos naturales gracias a la cooperación.

Aprovechando una de las convocatorias de INTERREG POCTEFA sumado a las ventajas de Copernicus, Gobierno de Navarra junto con otros organismos y empresas navarras y francesas presentó una propuesta de proyecto, denominado PyrenEOS y aprobado en 2016. La idea principal consiste en el desarrollo de una plataforma software y hardware que explote al máximo las imágenes para dar soporte a servicios de teledetección espacial ya existente y crear nuevos servicios destinados a la gestión de los recursos naturales de la zona. Gobierno de Navarra tiene como objetivo la creación de un repositorio de imágenes satélite en sus servidores con el fin de tener mayor accesibilidad para los usuarios y desarrolladores.

El presente Trabajo Fin de Grado se centra en la obtención de dichas imágenes de forma automática y su correspondiente adaptación al repositorio propio. Consistirá en una herramienta software opensource que no estará restringida para el uso de Gobierno de Navarra, si no que tenga la suficiente flexibilidad para ser usado por otras entidades. La motivación de este trabajo es el desarrollo de una aplicación que no se va a quedar apartada en una carpeta como un Trabajo de Fin de Grado, sino que se va a testear y poner en marcha en un entorno de producción real.

El documento se divide en tres partes: marco teórico (donde se resumirán conceptos que se han debido de tener en cuenta en el diseño del servicio), desarrollo del software (contendrá información detallada de cada bloque creado) y conclusiones con autocrítica del proyecto.

2. OBJETIVOS

Con el presente Trabajo Fin de Grado se pretenden cumplir los siguientes objetivos tanto en aspectos teóricos como prácticos:

- Adquirir conocimientos fundamentales sobre la teledetección espacial con el fin de evaluar el coste que supone la adquisición de una imagen y valorar el por qué Copernicus supone una revolución en materia de observación de la Tierra.
- Conocer a modo general el funcionamiento del programa Copernicus y, especialmente, de la familia de satélites Sentinel y los tipos de imágenes que cada plataforma ofrece.
- Obtener una visión global del proyecto PyrenEOS y otra más concreta del subsistema de descarga y creación de un repositorio propio para Gobierno de Navarra.
- Desarrollo de un software libre (opensource) de descarga y almacenamiento como proceso de servidor y no de usuario.
- Adecuación de dicho software al entorno tecnológico de Gobierno de Navarra, cumpliendo los requisitos que desde el punto de vista de Servicio de Infraestructuras son necesarios.
- Conseguir flexibilidad y autonomía de la herramienta. La primera característica tiene como objeto la facilidad de implementar nuevas opciones y de ser usado por otras entidades adaptándolo a su entorno de producción. El propósito de la segunda es dotar a la aplicación de la suficiente autonomía para ser capaz de tomar decisiones cuando una petición falla o supera un límite establecido, con soporte multiservidor.
- Desarrollo de interfaz de configuración y monitorización del sistema, con el fin de facilitar las tareas de soporte desde el puesto de trabajo del técnico responsable.



3. ESTADO DEL ARTE

3.1. FUNDAMENTOS DE LA TELEDETECCION ESPACIAL

Este apartado tiene como objetivo la introducción a conceptos básicos sobre la percepción remota (o teledetección) con el fin de comprender los datos que se van a obtener en la puesta en marcha del proyecto.

3.1.1. Teledetección espacial y sus aplicaciones

Técnica que permite la adquisición de imágenes de la superficie terrestre a distancia a través del análisis de los datos obtenidos por sensores instalados en satélites artificiales, gracias a la energía reflejada por los objetos que es capturada por el sensor y parcialmente procesada a bordo de éste [1].

La teledetección espacial involucra los siguientes elementos de acuerdo con la Figura 1:

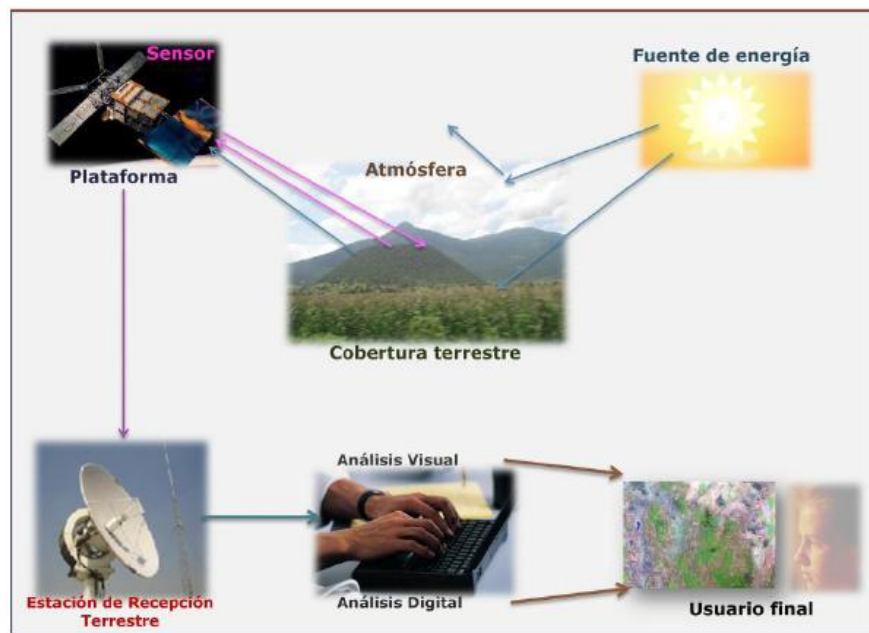


Figura 1 : Diagrama de los elementos del proceso de percepción remota o teledetección. Fuente: INEG [2]

- Fuente de energía o iluminación: radia energía electromagnética a los objetos terrestres de interés. En el siguiente apartado se verán los diferentes tipos.
- Atmósfera: ya que la energía emitida por el anterior interactúa con la atmósfera al viajar del sensor al objeto y viceversa. Esta interacción habrá que tenerla en cuenta para la corrección de imágenes.
- Objetos terrestres: son el objetivo de la observación. Dependiendo de sus propiedades, responderán a la energía recibida mediante la absorción y reflexión de ciertas longitudes de onda.

- Sensor remoto: encargado de recoger y grabar la radiación electromagnética reflejada o emitida por el objeto y la atmósfera. Este instrumento se encuentra instalado en plataformas, en nuestro caso satélites.
- Sistema de recepción y tratamiento de información: la energía grabada por el sensor se transmite a una estación receptora donde los datos se procesan y corrigen para convertirlos a imágenes digitales.
- Interpretación y análisis: consiste en interpretar la imagen, diseñando y desarrollando algoritmos eficientes para ello, con el fin de extraer la información geográfica de interés de los objetos captados.
- Usuario final: le da una aplicación a la información extraída en el proceso anterior [2]

Una de las líneas de investigación en percepción remota está asociada al uso y explotación de las imágenes de los satélites Landsat, SPOT y Sentinel para abordar la evolución de procesos territoriales relevantes para el desarrollo sustentable del país, monitoreo biofísico de los bosques, detección de fosas clandestinas, ... Por tanto, las imágenes obtenidas resultan esenciales para medir y monitorear tanto el clima como el medio ambiente de la Tierra, cómo se ve afectado el territorio y los procesos asociados al funcionamiento del planeta.

3.1.2. Tipos de resoluciones de un sistema

Los sistemas de teledetección se distinguen en la resolución, es decir, el nivel de detalle con que pueden capturar las imágenes, su frecuencia temporal, “finura espectral”, etc. Desde este punto de vista podemos considerar cuatro tipos de resolución que serán características importantes del sistema satélite del que proceden las imágenes a utilizar en este proyecto:

Resolución ESPACIAL:

La resolución espacial se refiere coloquialmente a la finura de detalles visibles en una imagen y está determinada por el tamaño del píxel, medido en metros sobre el terreno. Cuanto menor sea el área terrestre de cada píxel mayor resolución espacial pero también mayor el costo de las imágenes y la carga de proceso. Por ello, no siempre será necesario tener la mayor posible ya que la elección de una resolución suficiente dependerá del tamaño del objeto de interés.

Esta resolución depende de la altura del sensor con respecto a la Tierra, el ángulo de visión, la velocidad de escaneo y las características ópticas del sensor como por ejemplo su poder resolutivo. Además se deben considerar las influencias atmosféricas y los factores asociados a las características del objeto a detectar y de su entorno como pueden ser los niveles de iluminación, buenos contrastes y bordes nítidos.

Resolución ESPECTRAL:

El espectro electromagnético es el conjunto de todos los tipos de radiación electromagnética abarcando los rayos gamma, rayos x, luz ultravioleta, luz visible, luz infrarroja y ondas de radio. Cada tipo de radiación tiene una frecuencia y longitud de onda que la caracteriza como se puede ver en la Figura 2.

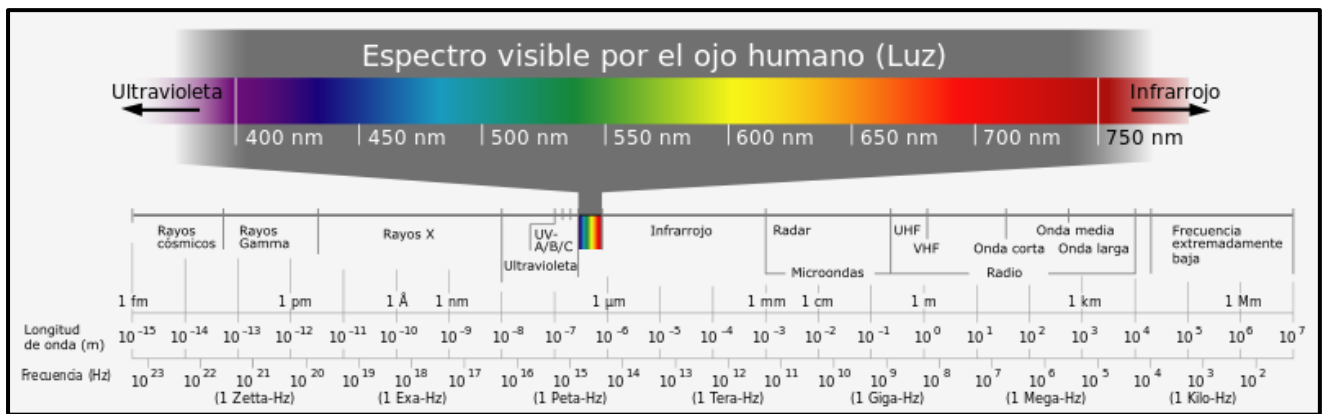


Figura 2: Características del espectro electromagnético. Fuente: Wikipedia [3]

Los sistemas de percepción remota están diseñados para captar imágenes en determinados rangos de longitudes de onda denominados bandas o canales. La resolución espectral se refiere al número y ancho de las bandas espectrales registradas por un sensor.

Si el sensor tiene la capacidad de registrar radiación electromagnética en varias longitudes de onda, se dice que es multiespectral. Si es capaz de registrar información en cientos de bandas se le llama hiperespectral. Dependiendo de la aplicación pueden seleccionarse sensores con bandas relativamente estrechas o anchas.

Las bandas en las imágenes se combinan para obtener información de la superficie de la tierra o de la atmósfera y para resaltar algunos rasgos de interés.

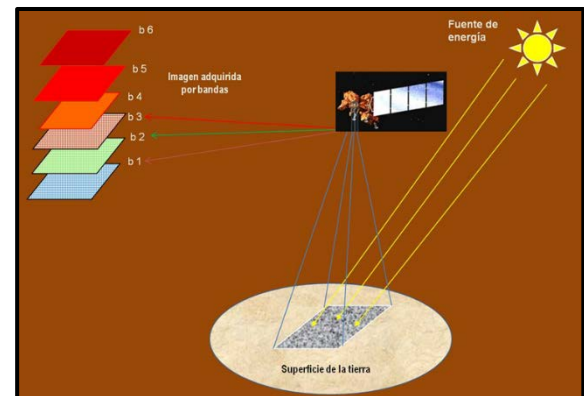


Figura 3: Concepto de bandas. Fuente: INEG

Resolución RADIOMÉTRICA:

La resolución o sensibilidad radiométrica se refiere al número de niveles de color en que se divide la radiación recibida para posteriormente ser almacenada y procesada. Cuanto mayor sea la cantidad de niveles, mayor será el detalle.

Resolución TEMPORAL:

Denominado también intervalo de revisita, se trata de la frecuencia de paso del satélite por una misma zona de la Tierra y depende prácticamente de las características de la órbita. En este caso una resolución alta, es decir un tiempo de revisita corto, es muy importante para el monitoreo de eventos que cambian en períodos cortos (inundaciones, incendios,...) y para el aumento de probabilidad de la obtención de imágenes “limpias” (sin nubes) en aquellas zonas de constante nubosidad. [2] y [4]

3.1.3. Obtención de una imagen satélite

Existen dos tipos de teledetección según la fuente de energía utilizada: pasiva (con sensores pasivos en el satélite) y activa (con sensores activos) cuya representación se visualiza en la Figura 4.

- **PASIVA:** La radiación electromagnética proviene del Sol e incide sobre los objetos de la superficie terrestre llegando al sensor la reflejada por estos últimos. También se habla de teledetección pasiva cuando la energía reflejada captada por el sensor proviene de la radiación terrestre emitida por los objetos (infrarrojo térmico)
- **ACTIVA:** Los sensores del satélite emiten su propio haz de energía siendo más independientes ya que no depende de las condiciones externas al sistema. [5]

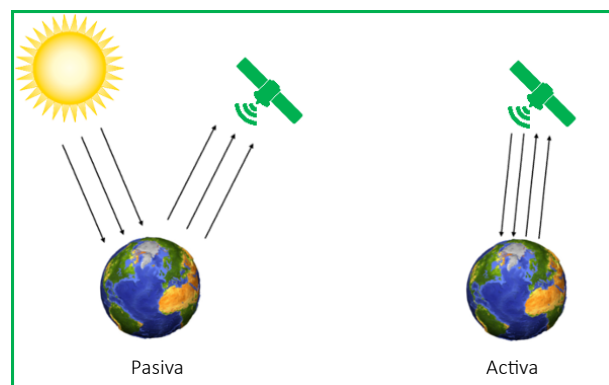


Figura 4: Representación teledetección pasiva y activa.
Fuente: GrindGIS [6]

Los objetos terrestres iluminados por el espectro fuente reflejan las longitudes de onda no absorbidas según la estructura y composición molecular de éstos como se verá en el siguiente apartado. Las radiaciones reflejadas se combinan generando “señales” de las que se puede extraer información acerca de dichos cuerpos y coberturas terrestres. Éstas son capturadas por los sensores del satélite, siendo parcialmente procesadas a bordo de éste creando una imagen satélite.

Una imagen digital está formada por un conjunto de píxeles ubicados en la intersección de cada fila i y columna j en cada una de las k bandas. Cada píxel en cada banda está en perfecto registro geométrico con sus equivalentes de las restantes bandas. Asociado a cada píxel existe un número digital (DN) que mide la radiación promedio o brillo correspondiente al área de escena abarcada por dicho píxel. Esta área puede incluir varios tipos de objetos dado que estamos hablando de resoluciones espaciales de metros o incluso kilómetros. La radiación reflejada correspondiente a dicho píxel que llega al sensor estará formada por las contribuciones de las firmas espectrales de los objetos de esa área.

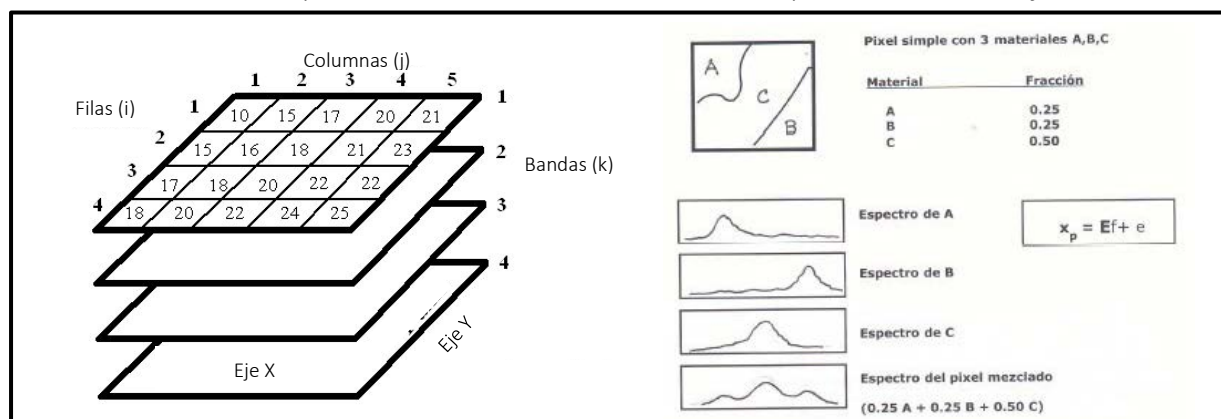


Figura 5: Composición de una imagen satélite y de un píxel con varios objetos. Fuente: Teledet [4]

La imagen de satélite resultante consiste en un conjunto de matrices, una por cada canal del sensor (por cada banda) con valores numéricos por cada píxel que serán necesario transformarlos en algún tipo de variable física. La transformación suele ser a través de una ecuación lineal cuyos parámetros se suministran junto con la imagen.

Sin embargo, la imagen obtenida contendrá posiblemente errores atmosféricos debido a que los efectos atmosféricos, como por ejemplo la niebla, contienen partículas que desvían los fotones de su trayectoria recta. Esto hace que el sensor registre energía proveniente de objetos que no están

ubicados en el campo de visión de los detectores del sensor. Este es un punto a tener en cuenta en el proceso de corrección de imágenes en un sistema de tratamiento. [4]

3.1.4. Respuesta espectral de la superficie

En este último punto se va a resumir la respuesta espectral de la superficie de los elementos más comunes que se encuentran en una imagen satélite. Con el fin de identificar objetos y procesos en la Tierra, la reflectancia de estos (radiación reenviada al espacio) respecto a las distintas longitudes de onda es el tipo de interacción que interesa. El gráfico que da la reflectividad (%) en cada longitud de onda se conoce como *signatura espectral* y constituye una marca de identidad de los objetos.

Cada tipo de material, suelo, vegetación, agua, etc. reflejará la radiación incidente de forma diferente lo que permitirá distinguirlo de los demás si medimos la radiación reflejada. Esta distinción se representa en la Figura 6.

El agua tiene una reflectividad muy baja absorbiendo casi toda la radiación que le llega en las bandas del infrarrojo próximo y medio, sin embargo un agua turbia aumentaría la reflectividad en el infrarrojo.

La vegetación tiene una reflectividad baja en el visible aunque con un pico en el color verde debido a la clorofila. La reflectividad es muy alta en el infrarrojo reflejado o próximo debido a la escasa absorción de energía por parte de las plantas en esta banda.

Finalmente el suelo tiene una reflectividad relativamente baja para todas las bandas aunque aumentando hacia el infrarrojo. Sin embargo va a depender mucho de la composición química y mineralógica, la textura y del contenido de humedad.

No solo es posible distinguir distintos elementos de una superficie, sino que también se puede diferenciar dentro de cada elemento, por ejemplo vegetación sana o seca, diferenciar un roble de un nogal, tierra árida o tierra húmeda, etc. Un ejemplo de ello se encuentra en la Figura 7, donde se compara la reflectancia entre una planta saludable y otras estresadas. Cuando el contenido de agua aumenta disminuye la reflectividad, ya que aumenta la absorción de radiación por parte del agua contenida en la planta. [4] y [5]

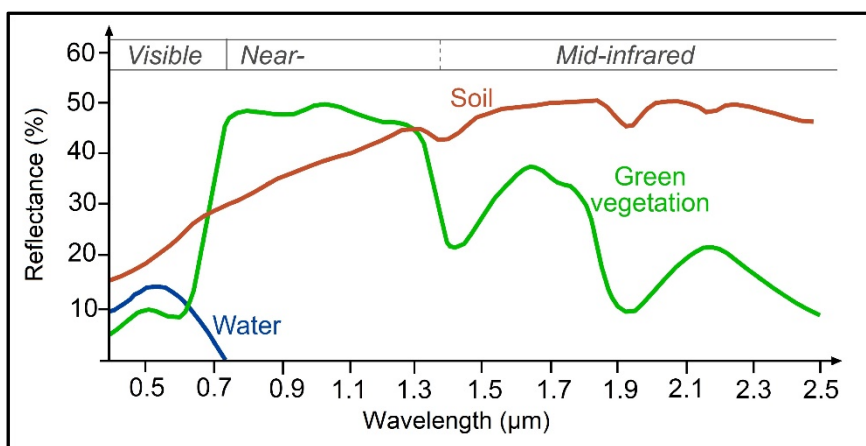


Figura 6: Comparación de la signatura espectral del agua, suelo y vegetación.

Fuente: SEOS (Science Education through Earth Observation for High Schools) [7]

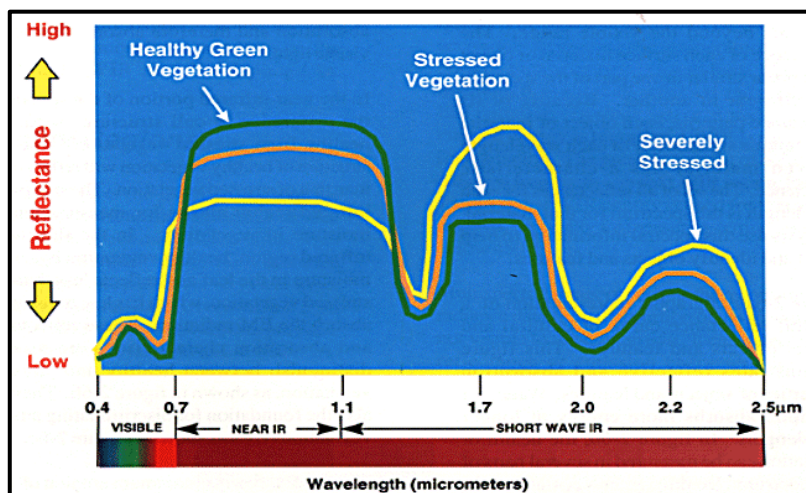


Figura 7: Comparativa de la reflectancia entre una planta sana y una estresada.

Fuente: NOAA (National Oceanic and Atmospheric Administration) [8]

3.2. PROGRAMA COPERNICUS

La presentación de este apartado tiene como objetivo la aportación de una visión general sobre el programa europeo Copernicus, siendo la fuente de los datos a utilizar en este Trabajo de Fin de Grado.

3.2.1. Copernicus en breve

Programa europeo liderado por la Comisión Europea e implementado en colaboración con los Estados Miembros de la Unión Europea, la Agencia Espacial Europea (ESA), la Organización Europea para la Explotación de Satélites Meteorológicos (EUMETSAT), el Centro Europeo de Previsiones Meteorológicas a Plazo Medio (ECMWF), entre otros.

Copernicus, anteriormente conocido como GMES, es el programa de la Unión Europea de observación y monitorización de la Tierra, que analiza el planeta y su medio ambiente en beneficio de los ciudadanos europeos. Está diseñado para proporcionar información precisa, actualizada y de fácil acceso para mejorar la gestión del medio ambiente, comprender y mitigar los efectos del cambio climático y garantizar la seguridad ciudadana.

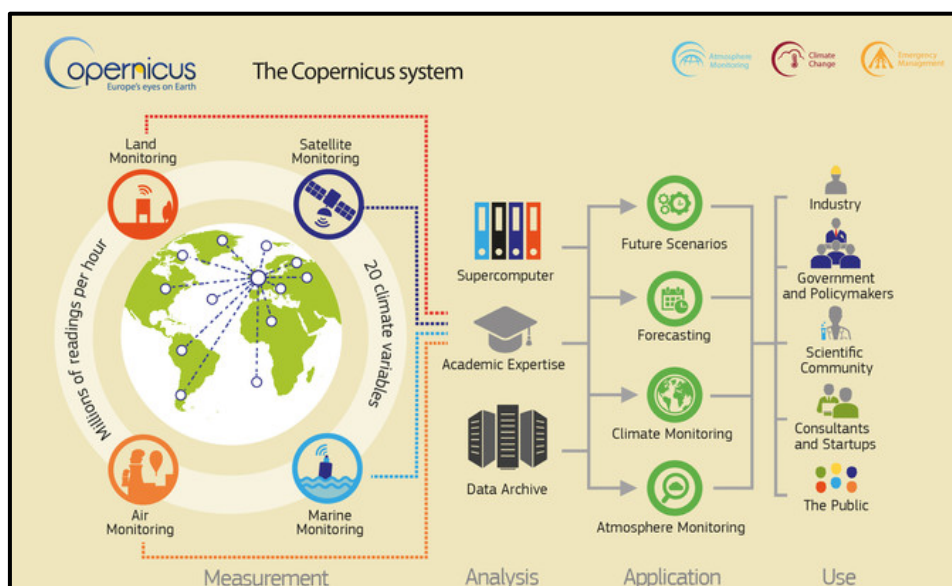


Figura 8: Visión general del sistema del programa Copernicus. Fuente: Copernicus [9]

El programa proporciona datos casi en tiempo real de manera operacional y servicios de información de forma gratuita sobre numerosas áreas de aplicación gracias a los datos recogidos por satélites en el espacio (algunos dedicados y otros procedentes de misiones contribuyentes) y por sistemas in-situ de medición en tierra, mar y aire adquiridos por multitud de sensores terrestres, marinos y aéreos. La ESA es la responsable de coordinar la distribución de los datos recogidos por más de 30 satélites, mientras que la EEA (European Environment Agency) gestiona el componente in-situ.

Los Servicios Copernicus transforman esta gran cantidad de datos satélite e in-situ en información de valor añadido mediante el procesamiento y análisis de los datos. Estos servicios se enmarcan en seis grandes categorías: gestión del suelo, el medio marino, la atmósfera, respuesta en situaciones de emergencia, seguridad y cambio climático. [10]

Sin embargo, muchas aplicaciones requerirán acceso a las imágenes no procesadas, es decir, a los datos que las estaciones terrestres de la ESA y EUMETSAT reciben de los satélites. Por ello, se han establecido diferentes infraestructuras de puntos de acceso de datos gratuito para que cualquier ciudadano u organización disponga y acceda fácilmente a dichas imágenes o, si lo desean, a los servicios anteriormente mencionados, como se representa en la Figura 9.

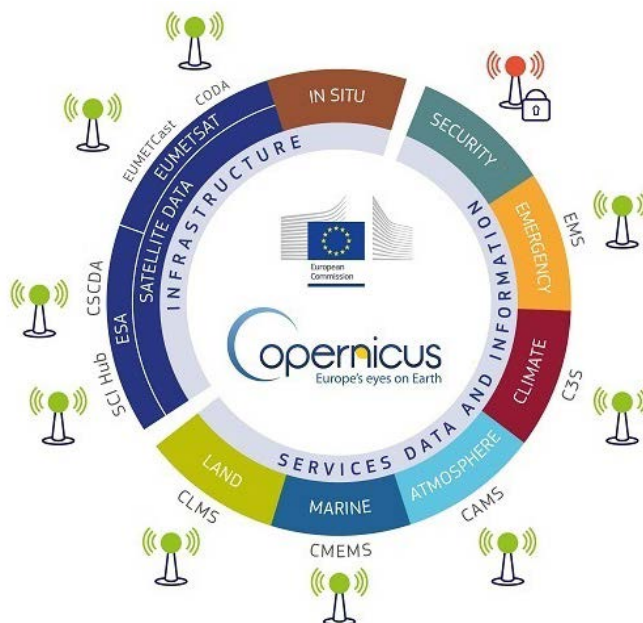


Figura 9: Puntos de acceso a los servicios Copernicus y a las infraestructuras de datos satélite. Fuente: Copernicus [9]

3.2.2. Misiones satélite: familia Sentinel

Copernicus se apoya en satélites destinados a la observación de la Tierra, clasificándose en 2 grupos: los satélites dedicados, explícitamente desarrollados para las necesidades específicas del programa, y las misiones contribuyentes (aproximadamente 30). Los primeros son una familia de satélites llamados Sentinel, ver Figura 10, que suponen una fuente constante e independiente de datos de alta calidad para los Servicios Copernicus. Cada misión Sentinel se compone de una constelación de 2 satélites iguales en la misma órbita pero con 180º de desfase, consiguiendo mayor cobertura y resolución temporal.

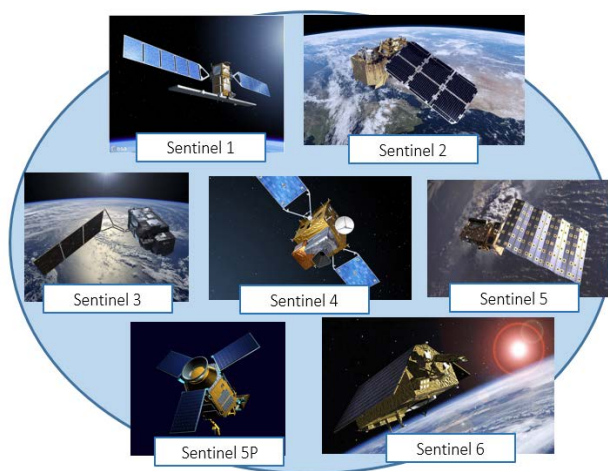


Figura 10: Familia de satélites Sentinel. Fuente: ESA [11]

La ESA es responsable de los datos de Sentinel 1, 2 y en futuro con el 5P, además proporciona los datos terrestres del Sentinel 3. Por otro lado, EUMETSAT se encarga de los Sentinel 4, 5 y 6 y de ofrecer los datos marítimos del 3. [10]

El programa Copernicus es un proyecto a largo plazo, dado que es muy reciente (llevando ya una duración de 3 años) no todos los satélites se encuentran en órbita actualmente. La planificación de los lanzamientos quedan reflejados en la Figura 11, presentando realmente algún ligero retraso.

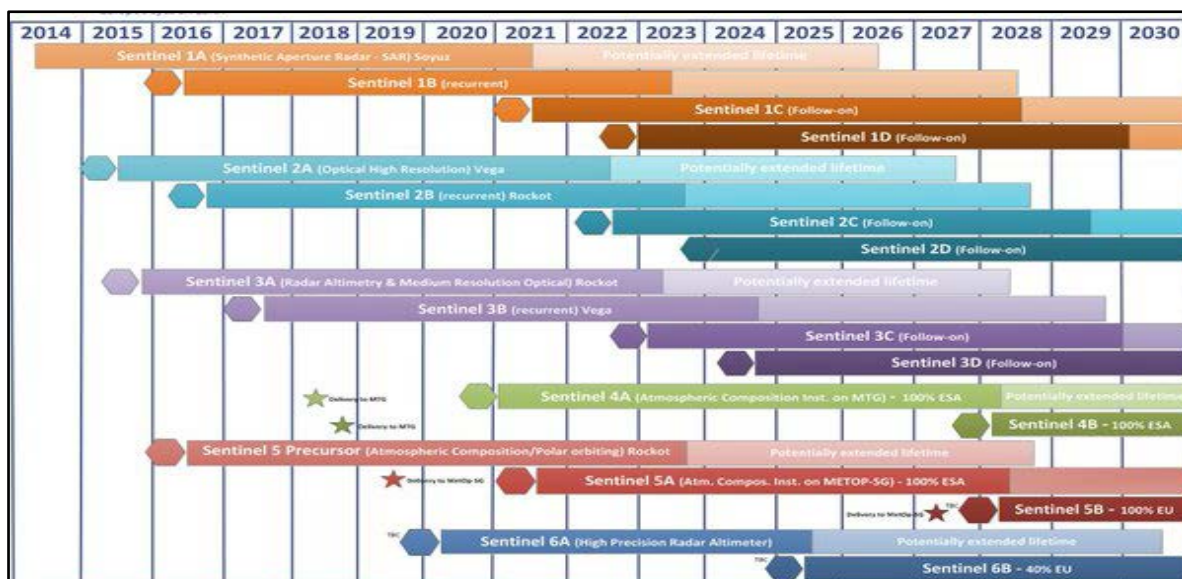


Figura 11: Programa de lanzamiento de satélites Sentinel. Fuente: Copernicus [9]

3.2.2.1. Sentinel 1

Es una misión radar en órbita polar capaz de tomar imágenes en cualquier condición meteorológica, de día o de noche las 24 horas gracias al radar, proporcionando datos de la superficie terrestre y los océanos. Su uso estará destinado a la observación de eventos ambientales, el apoyo a misiones de asistencia, rescate y emergencia durante desastres, entre otros. Dependiendo del modo de adquisición la resolución espacial es mayor o no, estando en un rango de los 5 a los 40 metros por pixel. El tiempo de revisita de cada satélite (o resolución temporal) es de 12 días, juntando ambos se consigue un intervalo de revisita de 6 días sobre un mismo punto. [11] y [12]

3.2.2.2. Sentinel 2

Consiste en una misión multispectral (abarca 13 de bandas del espectro electromagnético) de alta resolución en órbita polar, consiguiendo un tamaño de pixel de 10, 20 o 60 metros. Entre los dos satélites se obtiene una imagen del mismo punto cada 5 días. Está diseñada para monitorizar la tierra estudiando la cubierta vegetal, los usos del terreno y de las aguas, los cauces navegables y las zonas costeras, además de datos para los servicios de emergencias [11] y [12]

3.2.2.3. Sentinel 3

Transporta distintos instrumentos para medir la elevación, la temperatura y color superficial y de los océanos, y la topografía de estos, con gran precisión y fiabilidad. Se utiliza para medir el nivel de humedad presente en el recorrido de los pulsos de radar. Contribuirá a los servicios de pronóstico marítimo y de la monitorización del clima y del medio ambiente. Sus resoluciones dependen del tipo de instrumento, debido a que trasporta 4: OLCI, SLSTR, SRAL y MWR, siendo 300 metros la máxima resolución espacial posible, con un número considerable de bandas medidas. [11] y [12]

3.2.2.4. Sentinel 4

Instrumento compuesto por dos espectrómetros a bordo del satélite Meteosat Third Generation-Sounder (MTG-S) en una órbita geoestacionaria. Se concentrará en servicios tales como el análisis de la composición química de la atmósfera y el seguimiento de la calidad del aire sobre Europa y Norte de África, midiendo las concentraciones de aerosoles, gases traza y cubierta nubosa en la troposfera inferior. El objetivo es generar una parte cada hora sobre los datos de calidad del aire aunque su característica fundamental es el rango espectral que abarca (3 bandas). [12] y [13]

3.2.2.5. Sentinel 5

Instrumento embarcado en un satélite Metop de Segunda Generación que monitorizará también la atmósfera de nuestro planeta desde una órbita polar. Proveerá de datos de alta resolución y medidas precisas de los componentes atmosféricos como ozono, dióxido de nitrógeno, monóxido de carbono, etc. [12] y [13]

3.2.2.6. Sentinel 5 Precursor

Satélite concebido para reducir la discontinuidad de datos que existe entre las mediciones de las misiones de Envisat y Sentinel-5. Proporcionará observaciones de la química atmosférica, garantizando la continuidad de datos a medida que las misiones de seguimiento climático existentes llegan al fin de su vida. [11] y [12]

3.2.2.7. Sentinel 6

Ofrecerá altimetría de alta precisión para medir a nivel global la altura de la superficie marítima, principalmente pensado para oceanografía operacional y estudios climáticos. Colaborará con la ESA y EUMETSAT, NOAA y NASA. [12] y [13]

3.2.3. Productos Sentinel

Los datos ofrecidos por cada satélite Sentinel se llaman productos, diferenciados y nombrados por el nivel de procesamiento (existiendo tres: 0, 1 y 2), el modo de adquisición e incluso por la resolución. Claramente el número de productos dependerá del tipo de satélite, por ejemplo el Sentinel 2 tiene un único modo de captura mientras que el 1 tiene cuatro. Las Figuras 12, 13 y 14 representan estas distinciones y características de los distintos tipos y niveles de productos de aquellos satélites que se encuentran ya en órbita.

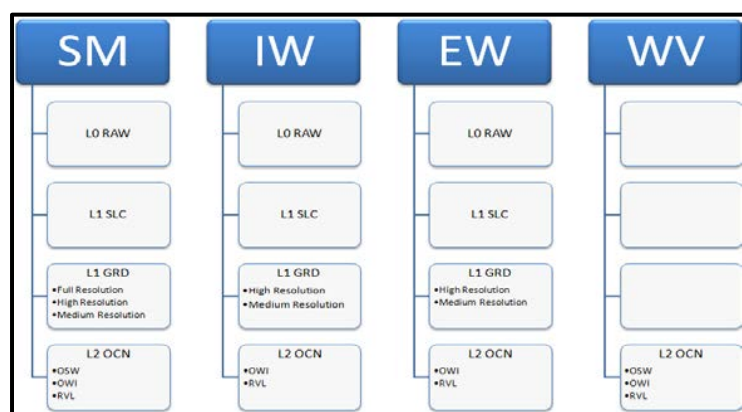


Figura 12: Esquema de niveles de productos por modos del Sentinel 1. Fuente: ESA [14]

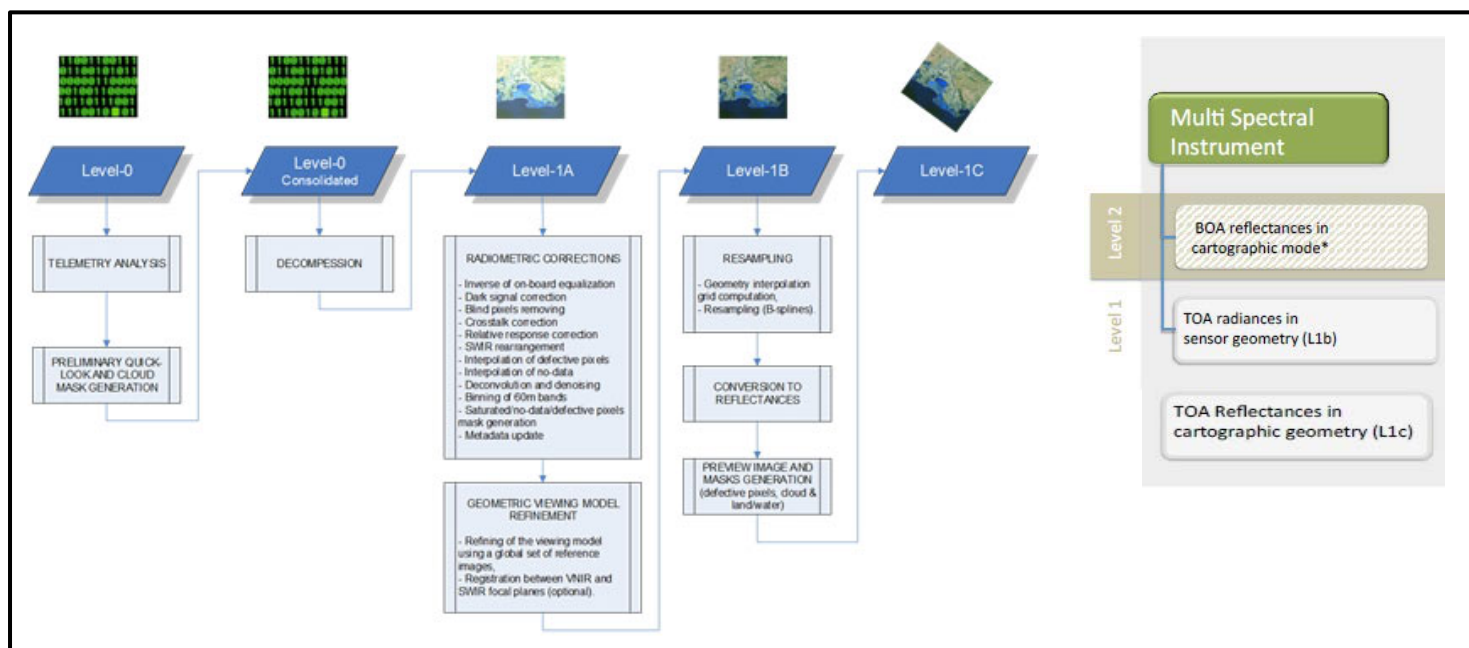


Figura 13: Esquema y características de niveles de productos del Sentinel 2. Fuente: ESA [15]

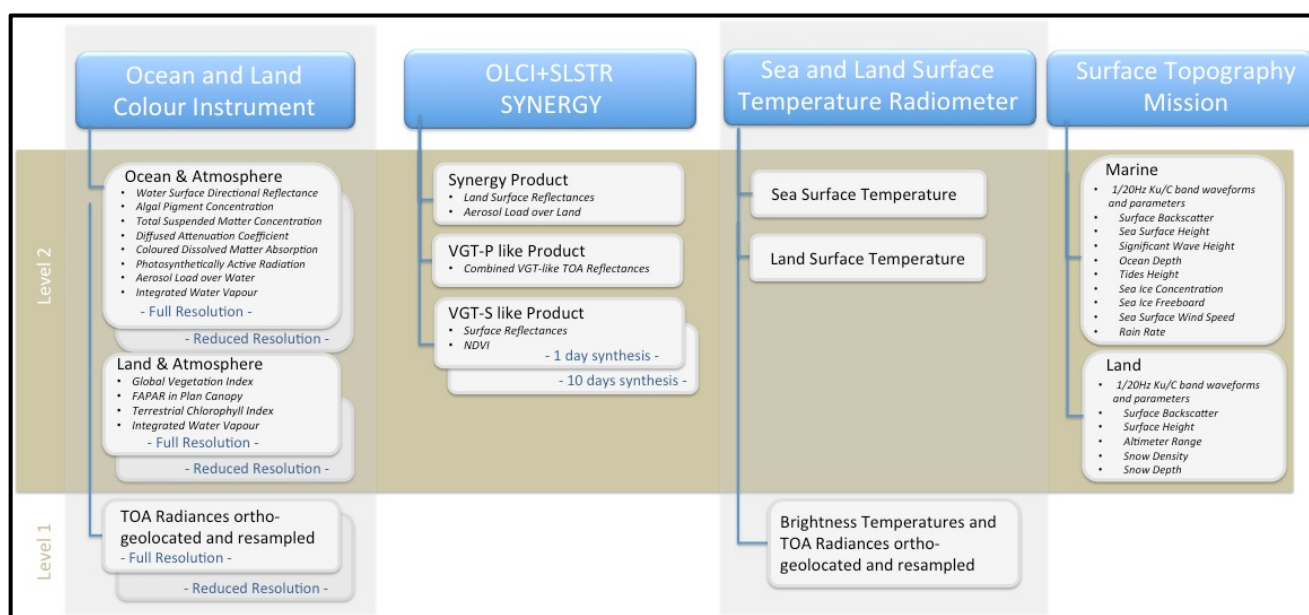


Figura 14: Esquema y características de niveles de productos por modos del Sentinel 3. Fuente: ESA [16]

Sin embargo, no todos los niveles se ponen a disposición de todos los usuarios: el nivel cero nunca es ofrecido por los puntos de acceso web a los datos, los accesibles son el nivel 1 (o 1C en el caso del Sentinel 2) y el 2, siendo más común el primero con el fin de tener la imagen más en bruto posible. Cabe destacar el proceso OLQC (Online Quality Control) que revisa la calidad de los productos L1C generados del Sentinel 2. [15]

Aun así, estimando que las necesidades de cada usuario son distintas, la ESA pone a disposición herramientas opensource para hacer el siguiente paso de preprocesamiento al nivel 2. Actualmente, se ha anunciado que hará accesible la descarga directa de esos productos de último nivel, únicamente los

que cubren Europa, con el fin de evitar mayores tiempos de espera que el preprocesado demoraba mediante dichas herramientas.

3.3. PROYECTO PYRENEOS

Este último apartado dentro del marco teórico tiene la finalidad de introducir a grandes rasgos el proyecto PyrenEOS (2017-2020), en el que se ha aportado con el desarrollo de la automatización de la descarga de imágenes Sentinel correspondiente a uno de los subsistemas del proyecto.

3.3.1. Visión general

El proyecto PyrenEOS consiste en el desarrollo de la primera plataforma transfronteriza pensada para generar nuevos servicios, mediante 5 herramientas software, orientados a un uso eficiente de los recursos naturales y a la gestión de riesgos naturales en los ámbitos de la gestión del agua, la gestión de cultivos agrícolas y las cubiertas forestales. Para ello, se basará en las capacidades de teledetección utilizando las imágenes satélites de Navarra y Pirineos Atlánticos ofrecidas por el programa Copernicus.

Debido a los retos tecnológicos que supone, el proyecto cuenta con 8 socios: Gobierno de Navarra, Département des Pyrénées-Atlantiques, Telespazio France, Tracasa, UPNA, INTIA, Fundación HAZI y CEREMA. Este proyecto fue definitivamente aprobado en Octubre de 2016 y comenzó en Marzo de 2017, desarrollándose durante los próximos tres años. [17]

3.3.2. Servicios

Como se ha explicado, PyrenEOS tiene su fin en el uso de 5 herramientas software a desarrollar, las cuales se clasifican en 2 tipos de servicios: VERDE, servicios piloto en gestión de cultivos agrícolas y cubiertas forestales, y AZUL, servicios en gestión del agua. En las Tablas 1 y 2, figuran más explícitamente las características de las herramientas a desarrollar por los distintos socios. [18]

Nº Herramienta	Nombre Herramienta	Beneficios	Socio desarrollador
1	Herramienta de verificación de declaraciones Política Agrícola Común (PAC)	Información a nivel de parcela catastral, relativa a la probabilidad de acuerdo entre el cultivo declarado por el agricultor y los posibles cultivos PAC	UPNA y Gobierno de Navarra
2	Servicios de ayuda a la decisión en la producción agrícola para la gestión de fertilización nitrogenada y el seguimiento de alertas en el desarrollo vegetativo.	Sistema de ayuda a la decisión en: fertilización nitrogenada en cultivos y sistema de seguimiento de alertas en el desarrollo vegetativo de cultivos a nivel de parcela.	UPNA e INTIA
3.a.	Herramienta para el seguimiento de anomalías en cubierta forestal	Identificación de anomalías en el desarrollo de las cobertura forestal y vegetal, consecuencia de accidentes bióticos o ligados a accidentes climáticos, como incendios	Telespazio y HAZI
3.b.	Herramienta para estimación de biomasa forestal y volumen de madera	Medición de volúmenes de biomasa, crecimientos y rangos de edad de desarrollo vegetativo en cubierta forestal	Telespazio y HAZI

Tabla 1: Herramientas de los servicios VERDE de la plataforma PyrenEOS.

Nº Herramienta	Nombre Herramienta	Beneficios	Socio desarrollador
1.a.	Cartografía de cobertura de nieve en alta resolución	Medición de nieve en superficie, información base para otros servicios y para operadores económicos relacionados con recurso nieve	Telespazio
1.b.	Herramienta de cálculo del recurso agua en manto níveo	Estimación de los volúmenes de agua almacenada en la nieve y hielo en las montañas	Telespazio
2.a.	Herramienta de cálculo de volumen agua en cursos y pantanos y detección de cambios en la morfología de los cursos de agua	Estimación de los volúmenes de agua en cursos de agua y pantanos y observación de cambios morfológicos en los cursos	Telespazio y CEREMA
2.b.	Herramienta de previsión y alerta de inundación	Simulación de la circulación del agua a través de valores simulados o a través de datos en tiempo real proporcionando un servicio de simulación de zonas inundables	Telespazio y CEREMA
2.c.	Atlas de Riesgos de inundación	Atlas con la identificación de zonas altamente inundables como resultados de simulaciones anteriores en base a diferentes condiciones meteorológicas	Telespazio y CEREMA

Tabla 2: Herramientas de los servicios AZUL de la plataforma PyrenEOS.

3.3.3. Subsistemas de la plataforma PyrenEOS

Para conseguir las aplicaciones del apartado anterior y con el fin de conseguir un sincronismo y cooperación entre los distintos socios se ha dividido el plan del proyecto en acciones:

- Acción Nº0: Preparación del proyecto
- Acción Nº1: Gestión del proyecto
- Acción Nº2: Actividades de comunicación
- Acción Nº3: Desarrollo de la plataforma PyrenEOS
- Acción Nº4: Desarrollo de herramientas piloto en gestión del agua – Herramienta AZUL
- Acción Nº5: Desarrollo de herramientas piloto en gestión agrícola y forestal – Herramienta VERDE
- Acción Nº6: Transferencia y explotación de la plataforma y herramientas piloto

El Trabajo de Fin de Grado presente se sitúa en la Acción Nº3 del proyecto, por lo que es de interés tener una visión general de esta etapa. La plataforma transfronteriza estará compuesta por tres subsistemas operativos:

1. ALMACENAMIENTO DE IMÁGENES: Consiste en los repositorios de información, el servidor que tiene acceso directo a ellos y el software que se encarga de aplicar políticas de copia, transferencia y acceso a las imágenes y datos necesarios para proporcionar los servicios a desarrollar. Para ello, Gobierno de Navarra proporcionará la infraestructura tecnológica más eficiente en su Centro de Procesado de Datos.

2. GEOPROCESO DE IMÁGENES: Se trata de los servidores de geoproceto que recibirán solicitudes de acceso según los estándares Open Geospatial Consortium (OGC) existentes y se encargará de acceder a los datos necesarios, hacer transformaciones y geoprocetos sobre las imágenes que se solicitan desde el subsistema 3.
3. INTEGRACIÓN Y ACCESO A PLATAFORMAS MÚLTIPLES: Este último consiste en los servidores y clientes encargados de mostrar la información procesada y llamar a los servicios existentes. [18]

De acuerdo con el título del documento y como se verá a lo largo de los siguientes apartados, se hace obvio que el programa desarrollado por el estudiante como Trabajo de Fin de Grado se centrará en el marco del primer subsistema.



4. SOFTWARE DE DESCARGA AUTOMÁTICA DE IMÁGENES SENTINEL

En este capítulo se cumplirán los objetivos marcados en el punto 2 con el desarrollo de una herramienta software de descarga adaptada al entorno tecnológico de Gobierno de Navarra con el fin de la creación de un repositorio propio, ofreciendo mayor rapidez de acceso a las imágenes. Para ello, en primera instancia se deberán conocer las necesidades de este organismo, junto con las de los socios del proyecto PyrenEOS con el fin de desarrollar una aplicación servidora lo más flexible posible. Con esto, se procederá al diseño y puesta en producción de dicha herramienta.

4.1. PLATAFORMA PYRENEOS, ENTORNO TECNOLÓGICO E INFRAESTRUCTURA DE GOBIERNO DE NAVARRA

4.1.1. Requisitos tecnológicos

El Servicio de Infraestructuras y Centro de Soporte de Gobierno de Navarra definió las siguientes pautas y requisitos a cumplir previo al diseño del software a través de un Estudio de Viabilidad:

- La solución no es una solución de usuario sino un proceso de servidor por lo que no debe tener dependencias con recursos específicos de usuario final, como es un navegador, variables de perfil de usuario etc.
- La plataforma de ejecución de procesos desatendidos está basada en el sistema operativo Windows.
- La solución puede estar basada en java, ejecutables de .net, scripts que utilicen comandos o aplicaciones que se puedan invocar desde un ejecutable sin instalar en el servidor
- La ejecución de la descarga deberá poder realizarse mediante un proceso en segundo plano que no requiera una sesión de usuario ejecutándose en el servidor.
- También sería conveniente que el sistema sea lo más autónomo posible, en el sentido de que pueda controlar de alguna forma que las descargas de imágenes se han realizado correctamente y ser capaz de lanzarlas de nuevo si fuera necesario.
- Con el fin de dotar facilidades a las tareas de soporte es necesario que el proceso deje log de lo que está haciendo.
- No hay definidos tiempos para recuperar en caso de avería, pero no tienen por qué ser bajos ni los de descarga ni los de almacenamiento
- Es un servicio que deberá actualizar diariamente las imágenes en las ventanas vacantes de los servidores. Las imágenes deberán estar disponibles en el repositorio para los servicios permanentemente.
- Los escenarios de acceso serán Internet para el servicio de descarga y la Red Interna de Gobierno de Navarra para los responsables de comprobación de estado del repositorio propio.

4.1.2. Plataforma software diseñada y Escenario hardware planteado

Como se ha explicado en el apartado 3.3., PyrenEOS es una plataforma transfronteriza. Sin embargo, tras periódicas reuniones se acordó que cada una de las regiones diseñaría y desarrollaría su propio sistema con la consecuente creación de dos catálogos de imágenes, siendo el punto de unión la plataforma de los usuarios finales. La arquitectura general del software correspondiente a Navarra se representa en la Figura 15, diseñada por Álvaro Huarte, desarrollador de software GIS en Tracasa, donde se distinguen claramente los 3 subsistemas principales.

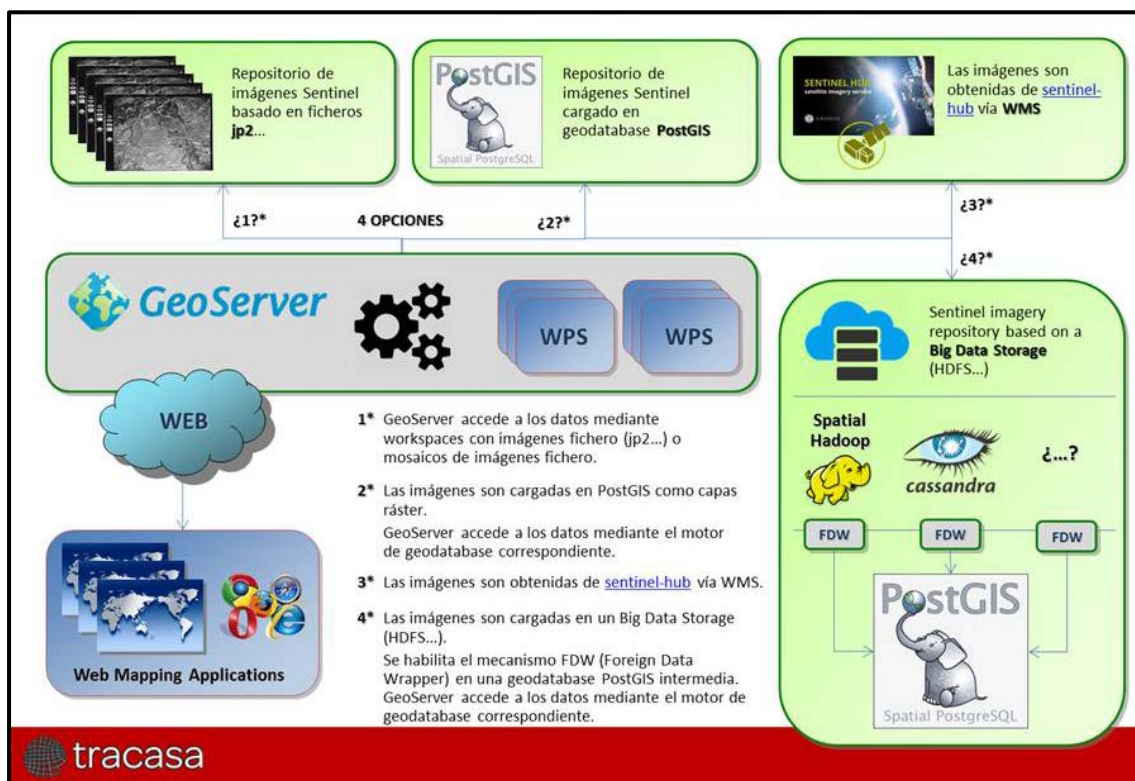


Figura 15: Arquitectura software propuesta para la plataforma navarra PyrenEOS. Fuente: Álvaro Huarte. Tracasa [19]

Las incógnitas representadas en la Figura 15 en la sección del subsistema de descarga son con motivo de la gran cantidad de información a almacenar estimada, siendo previsible la evolución de un repositorio con todas las imágenes basada en ficheros, pasando por su organización mediante bases de datos geográficas (PostGIS [20]) y concluyendo en uno con mayor refinamiento y complejidad utilizando técnicas Big Data. Este Trabajo Fin de Grado se centrará en las opciones 1 y 2 planteadas.

Debido a la incertidumbre ante la tipología de productos Sentinel requeridos por los desarrolladores de las herramientas PyrenEOS, se realizó el cálculo de la cantidad media estimada en un año a almacenar en el CPD de Gobierno de Navarra en el caso peor: todos los tipos de productos (adquisiciones de distintos instrumentos y diferentes niveles de procesamiento) de los 6 satélites Sentinel actualmente en órbita con imágenes disponibles de todas las órbitas relativas por las que se captura Navarra o zonas de esta.

En la reunión realizada el 9 de Mayo de 2017 con el Servicio de Infraestructuras y Centro de Soporte, se llegó a la conclusión que los .zip descargados del servidor se debían mantener a modo de salvaguarda, con el fin de aprovechar posibles herramientas futuras que ESA o EUMETSAT saquen a la luz utilizando dichos archivos originales. Sin embargo, dado que el segundo subsistema (GeoServer [21] en la Figura 15), no requerirá de esos datos comprimidos, se determinó que estos ficheros estuvieran almacenados en una cabina de almacenamiento de lectura lenta, abaratando costes. El resto de productos (descomprimidos y pre-procesados) permanecerán guardados en cabina de lectura rápida para conseguir ligereza en el sistema. Por este motivo, se realizan en las Tablas 3, 4, 5 y 6 dos distinciones: la capacidad total en almacenamiento lento y total en almacenamiento rápido.

Productos	Tamaño ZIP original (GB)	Tamaño original (GB)	Imágenes anuales	Total almacenamiento lento (GB)	Total almacenamiento rápido(GB)
RAW -L0	4,5	7,5	487	2191,5	3652,5
SLC -L1	0,972	1,62	487	473,364	788,94
GRDH -L1	0,9	1,5	487	438,3	730,5
OCN -L2	0,00732	0,0122	365	2,6718	4,453

TOTALES	1826	3,106 TB	5,176 TB
----------------	-------------	-----------------	-----------------

Tabla 3: Cantidad media de imágenes y capacidad requerida anual de Sentinel 1A y 1B

Productos	Tamaño ZIP original (MB)	Tamaño original (MB)	Tamaño pre-procesado L2A (MB)	Imágenes anuales	Total almacenamiento lento (GB)	Total almacenamiento rápido(GB)
MSI-L1C	800	800	1600	402	321,6	964,8

TOTALES con 2 satélites	804	0,643 TB	1,93 TB
--------------------------------	------------	-----------------	----------------

Tabla 4: Cantidad media de imágenes y capacidad requerida anual de Sentinel 2A

En los totales de la Tabla 4 se consideran los dos satélites de la misión Sentinel 2 (A y B) ya que el lanzamiento del segundo se realizó con éxito el 7 de Marzo de 2017 y es cuestión de poco tiempo que sus datos estén disponibles y accesibles gratuitamente. En cuanto a las imágenes del satélite Sentinel-3A, se realizan dos distinciones: productos con información terrestre ofrecidos por ESA (Tabla 5) y con información oceánica por EUMETSAT (Tabla 6).

Productos	Nivel de productos	Tamaño ZIP original (MB)	Tamaño original (MB)	Imágenes anuales	Total almacenamiento lento (GB)	Total almacenamiento rápido (GB)
SAR	SR_2_WAT..._ST	97	97	41	3,977	3,977
	SR_1_SRA..._ST	51	51	41	2,091	2,091
	SR_2_WAT..._NR	6	6	41	0,246	0,246
	SR_1_SRA..._NR	10	10	41	0,41	0,41



OLCI	OLCI_ERR..._NR	710	710	216	153,36	153,36
	OLCI_ERR..._NT	710	710	216	153,36	153,36
	OLCI_EFR..._NR	665	665	230	152,95	152,95
	OLCI_EFR..._NT	665	665	230	152,95	152,95
SLSTR	SL_RBT..._NR	480	480	162	77,76	77,76
	SL_RBT..._NT	500	500	162	81	81
TOTALES			1380		0,778 TB	0,778 TB

Tabla 5: Cantidad media de imágenes y capacidad requerida anual de Sentinel 3A-LAND

Productos	Nivel de productos	Tamaño ZIP original (MB)	Tamaño original (MB)	Imágenes anuales	Total almacenamiento lento (GB)	Total almacenamiento rápido (GB)
SAR	SR_2_LAN	90	90	41	3,69	3,69
	SR_1_SRA_A	2200	2200	41	90,2	90,2
	SR_1_SRA_BS	1690	1690	41	69,29	69,29
	SR_1_SRA_	51	51	41	2,091	2,091
OLCI	OLCI_ERR..._LN1	700	700	216	151,2	151,2
	OLCI_ERR..._SVL	700	700	216	151,2	151,2
	OLCI_EFR..._LN1	670	670	230	154,1	154,1
	OLCI_EFR..._SVL	670	670	230	154,1	154,1
SLSTR	SL_RBT..._LN2	480	480	162	77,76	77,76
	SL_RBT..._SVL	480	480	162	77,76	77,76
TOTALES			1380		0,931 TB	0,931 TB

Tabla 6: Cantidad media de imágenes y capacidad requerida anual de Sentinel 3A-OCEAN

Sumando los totales de los 6 satélites, se calcula que la capacidad necesaria en el CPD es de 5,458 TB en cabina de almacenamiento de lectura lenta y 8,815 TB en cabina de almacenamiento de lectura rápida. Estos datos se realizan sin tener en cuenta las imágenes geo-preprocesadas como puede ser NDVI (valores de nitrógeno), optimización de las bandas (formato .tiff), etc., lo cual incrementaría dicho requerimiento, pero de nuevo es una incógnita no resuelta durante este periodo.

Con esta gran cantidad de datos se hace necesaria una arquitectura hardware eficiente para su manejo y transporte.

El diseño general que se le dotaría al sistema estaría compuesto por una red de servicio y administración y por otra red de almacenamiento:

- La infraestructura servidora del proyecto está basada en un sistema de virtualización VMWare que aporta o... la solución está soportada por servidores HP BL... HP C7000.



- El almacenamiento del proyecto está ubicado en una cabina IBM Storwize V5000, con 2 controladoras y con discos SSD de alta capacidad, que aportan un alto rendimiento para el procesamiento de las imágenes. Dicho sistema de almacenamiento es capaz de albergar hasta 504 discos con una capacidad total de casi 500 TeraBytes con la configuración actual. Esta capacidad se podría incrementar agregando más cabinas IBM Storwize V5000 a la solución o bien con la capacidad del sistema para comprimir la información.
- La conectividad entre la parte servidora y la parte del almacenamiento se realiza por protocolo Fiber Channel, a través de una red SAN (especializada en almacenamiento), con 2 fabrics que aportan disponibilidad a la solución
- La conectividad a nivel de red de almacenamiento (SAN) es de 8 Gbps, mientras que la de servicio (Ethernet) es de 10 Gbps.

Refinando esta visión general con el diseño acordado en la reunión mencionada, el escenario planteado se correspondería con la Figura 16:

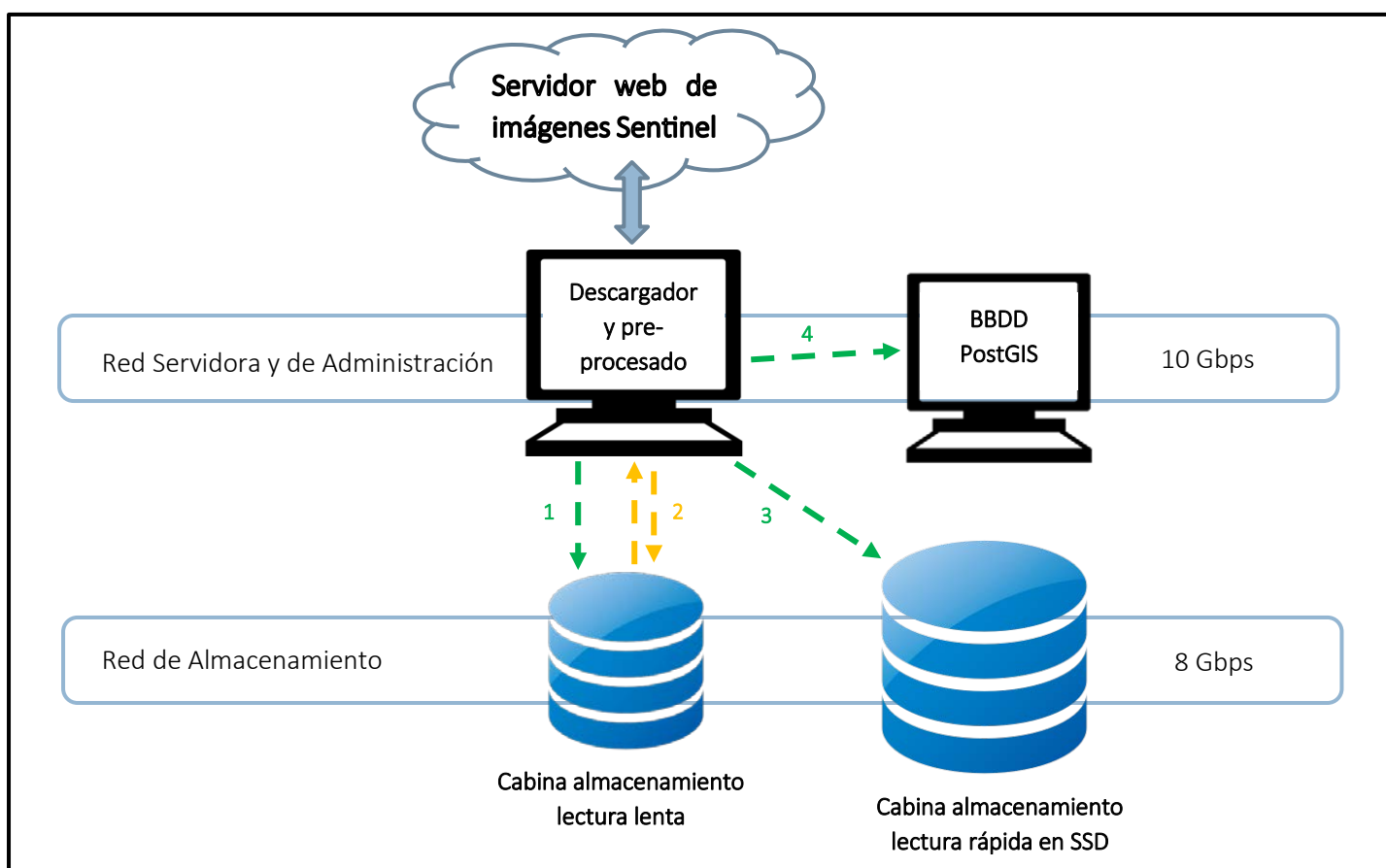


Figura 16: Escenario diseñado para la descarga y almacenamiento del proyecto PyrenEOS

Como se puede observar, el software de descarga se encuentra en una máquina dedicada al igual que la base de datos que se encuentra en otro servidor dedicado. El funcionamiento del software sería de acuerdo con la Figura 16:

1. El fichero descargado del repositorio web se guarda directamente en cabina lenta, destinado a los zips.

2. Una vez almacenado, el software pasa a comprobar su correcta descarga (no hay datos corruptos ni errores en las imágenes). Si se da uno de esos casos, elimina dicho fichero almacenado.
3. En caso que el archivo supere el control del paso 2, se descomprime y guarda en el almacenamiento rápido.
4. Por último, se introducen en las tablas correspondientes de la base de datos algunas de las propiedades geográficas del fichero de metadatos de la imagen en cuestión. Esta base de datos será la conexión entre el subsistema de descarga y almacenamiento y el subsistema GeoServer.

4.1.3. Escenario simulado

Actualmente, PyrenEOS se encuentra en su fase inicial de desarrollo. Es por esto que en el periodo en el que se ha elaborado este trabajo no se han tenido disponibles los recursos físicos planteados en el apartado anterior. Ante esta situación, se ha tomado la decisión de simular el entorno de producción, en la Red Interna de Gobierno de Navarra, en un ordenador personal de la siguiente manera:

- La máquina local representará el servidor de descarga y pre-procesado
- Se creará una máquina virtual con sistema operativo Windows y adaptador en modo Host Only, emulando el puesto de trabajo (ordenador o tablet) del técnico responsable de las tareas de monitorización.
- En una primera prueba, la base de datos y servidor web estarán en la propia máquina local. Una vez testado, se creará otra máquina virtual Windows (de nuevo en Host Only) recreando ser un servidor dedicado para la base de datos y servicio web.
- El almacenamiento rápido estará representado por un disco duro de 750 GB de capacidad conectado a un puerto USB 3.0, transfiriendo los datos a una velocidad teórica de hasta 4.8 Gbps.
- Por otro lado, el almacenamiento de lectura lenta se guardará en un dispositivo USB de 32 GB conectado a un puerto USB 2.0 de hasta 480 Mbps de velocidad, algo inferior en la práctica, siendo 10 veces más lento que el anterior.

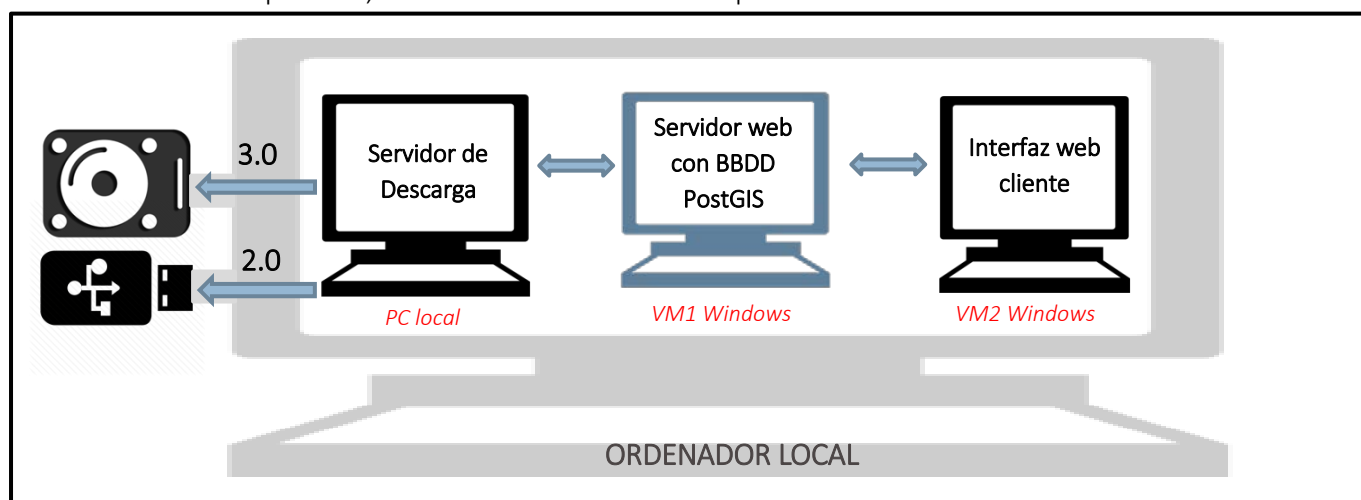


Figura 17: Recreación del escenario servidor-almacenamiento a simular

Aunque a priori pueda parecer poco eficiente, con esta metodología y consiguiendo una aplicación flexible, como se ha marcado en uno de los objetivos, se pueden simular varios escenarios de diferentes infraestructuras servicio-almacenamiento. De esta forma, no queda restringido al planteamiento inicial y no es susceptible a los más que previsibles cambios en topología.

4.2. ESTUDIO COMPARATIVO DE REPOSITORIOS WEB DE PROVEEDORES


4.2.1. Necesidad

El punto de partida para lograr un repositorio de imágenes propio es conocer desde qué servidor web descargar las imágenes a almacenar. Recordando la Figura 9, en el apartado 3.2.1, ESA y EUMETSAT ofrecen 4 puntos de acceso oficiales de manera gratuita. Los correspondientes a la primera entidad son SCIHUB y CSCDA, teniendo este último diferentes niveles de acceso por lo que SCIHUB [22] es el más utilizado a nivel global. CODA [23] y EUMETCast son los desarrollados por EUMETSAT, siendo más frecuente el primero debido a que EUMETCast no es un interfaz de descarga, si no que los productos se reciben en streaming con previo acuerdo de suscripción, requiriendo un software y un dispositivo específico.

Sin embargo y pese a ser servidores oficiales, constan de diversos problemas, principalmente SCIHUB cuyo conjunto de datos almacenado es considerablemente mayor a CODA en la actualidad:

- Disponibilidad: para hacerse una idea, diariamente este servidor publica en torno a 6000 productos y aproximadamente recibe 64.000 peticiones de descarga de imágenes, lo que requiere un gran soporte técnico. Es por esto que sufre a lo largo de un mismo mes numerosas tareas de mantenimiento, algunas de ellas planificadas y otras de manera imprevista, como se ejemplifica en la Figura 18. Esto se traduce en la indisponibilidad de un servicio 24x7 al 100%.

Maintenance activities on Data Hub Services starting from 4 April 2017 to 12 April 2017




A series of maintenance activities both at infrastructure and at application level are scheduled for the following dates:

- **4 April 2017** from 9:30 CEST to 16:30 CEST: *Open Access Hub* only. During the maintenance window users will not be able to perform downloads and access the service. API Hub will not be affected.
- **5 April 2017** from 9:30 CEST to 16:30 CEST: *All services* - During the maintenance window there will be a products publication delay for all missions on all services. No Downtimes are foreseen.
- **12 April 2017** from 9:30 CEST to 18:00 CEST: *All services* - During the maintenance window there will be occasional service downtimes and a delay on products publication.

We apologise for the inconvenience that this might cause.

Sentinel-2A products temporary unavailability



Due to a Sentinel-2 ground segment anomaly, the publication of Sentinel-2A products in the data hubs was temporarily unavailable on 7 April 2017 between sensing orbit 9345 and 9357.

The recovery is ongoing and is expected to be complete in the coming days, a corresponding news update will be published accordingly.

We apologize for the inconvenience this may cause.

12 Apr 2017 - 10:01

Figura 18: Ejemplos servicios de mantenimiento en SCIHUB. Fuente: Copernicus Open Access Hub [24]

- Saturación: el hecho de ser un repositorio oficial, convierte a SCIHUB en el epicentro de las descargas por parte de usuarios a nivel mundial, pudiendo llegar a saturar el servidor o hacer que su velocidad de salida sea inferior. Teniendo en cuenta los tamaños de las imágenes, como ilustraban las Tablas 3, 4 y 5, una velocidad relativamente baja se traduciría en un tiempo mayor de espera, lo que en servicios en caliente implicaría una experiencia negativa a nivel de usuario.
- Publicación de imágenes corruptas: de acuerdo con la explicación del apartado 3.2.3, las capturas procedentes de los satélites Sentinel 2, pasan por un proceso de control de calidad (OLQC). Sin embargo, sea cual sea el resultado, ESA pone a disposición dichas imágenes aunque son corregidas y subidas en un largo periodo pero sin seguir un patrón aparente. Un producto que falla en el control de niveles de calidad, puede implicar pérdida de datos, como se representa en la Figura 19, traduciéndose en el pequeño arcoíris y siendo una imagen no válida. Esto implica la posible descarga de imágenes “corruptas” y la necesidad de realizar una segunda descarga, aumentando el tiempo invertido.



Figura 19: Imagen no válida del satélite Sentinel 2A con pérdida de datos.

Gracias a la gratuidad de las imágenes aparecen distintos proveedores de estas, la mayoría de ellos dan acceso igualmente gratuito. Debido a las problemáticas comentadas en el apartado anterior, surge la idea en el proyecto PyrenEOS de aprovechar la disponibilidad de los nuevos servidores con el fin de tener un repositorio de backup desde el que descargar o incluso crear una plataforma multiservidor. Es así que se hace necesario un estudio comparativo de estos con el objetivo de tener como distribuidores los más eficientes posibles.

4.2.2. Puntos de Acceso Web y Tipología

En primer lugar, se ha tenido que investigar los servidores disponibles actualmente y sus ventajas y desventajas a priori. Para ello, se realizó una clasificación inicial de acuerdo a la tipología de cada una, teniendo 5 tipos de clases como muestra la Tabla 7.

Oficiales	Mirrors Nacionales	Mirrors Parciales	Iniciativas Privadas	Proveedores de Servicios en la Nube
SCIHUB CODA	AUSTRALIA N.M. AUSTRIA N.M FINLAND N.M PEPS (FRANCE N.M.) CODE-DE (GERMAN N.M) HNSDMS (GRECE N.M) ITALY N.M. PORTUGAL N.M SWEA (SWEDEN N.M) SEDAS (UNITED KINGDOM N.M)	ASF (Alaska Satellite Facility) THEIA USGS Earth Explorer CEDA (Centre for Environmental Data Analysis)	SENTINEL-HUB PLANET PLATFORM	AMAZON WEB SERVICES GOOGLE CLOUD PLATFORM EO CLOUD FERRO

Tabla 7: Listado y Clasificación de los Puntos de Acceso a imágenes Sentinel existentes

Pese a que los nombres de las clases sean intuitivos, a fin de comprender esta clasificación, se presenta a continuación un breve resumen de cada una de ellas, a excepción de los distribuidores oficiales que ya se ha introducido en el apartado anterior:

MIRRORS NACIONALES

Como la propia palabra inglesa indica, los mirror son espejos de los servidores de imágenes oficiales de la ESA, sin embargo el hecho de ser una copia no implica que estén afectados por los mismos problemas que SCIHUB ya que cada uno tiene sus propias labores de mantenimiento.

Su existencia es debida al desarrollo y acuerdo entre la ESA y los países miembro de la Unión Europea de los “segmentos terrestres colaborativos” (Collaborative Ground Segment), con el fin de permitir a usuarios, industrias o entidades de un área geográfica concreta un acceso a los datos satélite complementario al oficial. Estos segmentos cuentan con centros de procesamiento y archivo de datos, además de aplicaciones específicas para cada país y algunas de ellas disponen de estaciones de recepción y otras reciben los datos por difusión masiva desde la ESA. Pueden contener colecciones completas o parciales de acuerdo a selecciones geográficas.

La lista de estos espejos, reflejada en la Tabla 7, se encuentra en constante incremento debido al proceso de negociaciones que se están dando, destacando la más que posible creación en poco tiempo del mirror correspondiente a España. [25]

MIRRORS PARCIALES

Consisten en iniciativas para integrar datos Sentinel específicos en las infraestructuras de plataformas ya existentes con datos procedentes de otras misiones. La adquisición de las imágenes del programa Copernicus por parte de estos, es debida al acuerdo de cooperación internacional que la Comisión Europea lidera con países y organizaciones internacionales. Los posibles ámbitos de colaboración son en resumen la adquisición, producción en casi tiempo real y distribución de datos, además del desarrollo de herramientas y aplicaciones innovadoras de forma pública.

INICIATIVAS PRIVADAS Y PROVEEDORES DE SERVICIOS EN LA NUBE

A diferencia de los tipos anteriores, estos no tienen acuerdo con la ESA para la adquisición de datos, si no que desarrollan sus propios algoritmos para la obtención, filtrado y clasificación de las imágenes. Algunos de ellos realizan incluso pre-procesados y correcciones geográficas, mientras que otros ponen a disposición herramientas e interfaces de fácil uso para que un usuario final realice las correcciones que sean de su interés.

4.2.3. Requisitos a cumplir y primeros descartes

Con el fin de seguir unos criterios, en primera instancia, los socios de PyrenEOS establecieron una serie de requisitos que los servidores de imágenes, destinados a ser los proveedores de datos del proyecto, debían cumplir:

- Acceso a imágenes de forma gratuita para cualquier tipo de usuario
- Disponibilidad del producto (quasi-) Near Real Time (NRT – en tiempo real)
- Mantener la estructura del producto, es decir, nombres y organización de archivos y carpetas, sin la supresión de ninguno de estos o de incluso metadatos relevantes. Es importante conservar todos los elementos tal cual son recibidos por la ESA, como refleja el ejemplo en la Figura 20, ya que muchas de las herramientas de procesamiento y visualización ya existentes y futuras utilizan el producto en su totalidad.

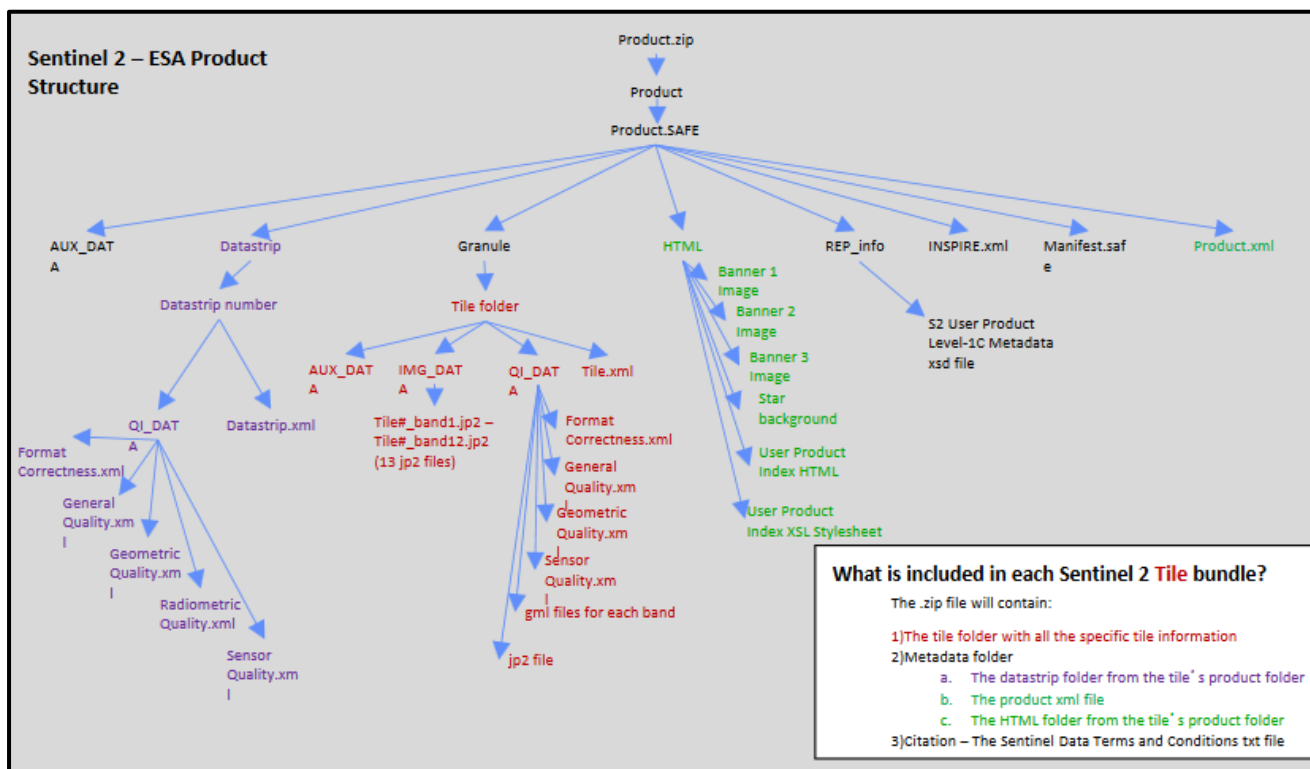


Figura 20: Estructura de contenidos de productos Sentinel 2. Fuente: USGS [26]

- Fiabilidad del repositorio, es decir, asegurarse de la existencia del servidor en un futuro.

Siguiendo estas condiciones, y previo a realizar el desarrollo de las herramientas de estudio, se logran descartar los distribuidores que se indican a continuación.

SENTINEL-HUB

Son varios los motivos por los que se prescinde de este repositorio, siendo el principal de ellos el hecho de necesitar un nivel de usuario con privilegios con el fin de acceder a los datos a través de su servicio, el cual es de pago por lo que no cumplirían el primer requisito.

Este servidor dispone de las imágenes pre-procesadas y nuevos productos generados a partir de la combinación de bandas. Sin embargo, la forma de acceder a estos datos no es una descarga convencional, si no que se realiza a través del estándar WMS (servicio web de mapas por el que se producen mapas de datos referenciados espacialmente dinámicamente a partir de información geográfica), entre otros parecidos. Basándose en la arquitectura software de la

plataforma planteada, Figura 15, se puede comprobar que este servidor quedaba contemplado en la tercera opción del subsistema 1 y que el subsistema 2 obtendría las imágenes pedidas por el usuario final a través de ese servicio. Sin embargo, esto implicaría que los productos no quedarían almacenados y por tanto la idea de un repositorio propio no tendría sentido en este caso. [27]

USGS EARTH EXPLORER

El presente repositorio no es buen candidato a ser nuestro proveedor debido a que la manera de descarga de datos es mediante navegador. Si bien es cierto que se ofrece un interfaz de descarga por diseminación masiva a instalar en el ordenador, la selección de los productos es obligatoriamente por un navegador web, siendo la solución del proyecto PyrenEOS una descarga en el lado del servidor. No se pone a disposición de usuario ningún tipo de API que seguir con el fin de desarrollar scripts para la ingestión automática de imágenes. [26]

CEDA

La razón de descarte de este proveedor es muy similar al anterior. Pone a disposición de manera gratuita y abierta un interfaz opensearch y su respectivo código a través de su cuenta oficial en Github. Sin embargo, de nuevo es un interfaz de cliente y no ofrece ninguna API para lograr la automatización de descargas desde un servidor. [28] y [29]

PLANET PLATFORM

A diferencia de CEDA y USGS EARTH EXPLORER, esta plataforma además de tener interfaz de usuario también cuenta con una API explicada al detalle para la búsqueda y descarga de los datos requeridos. Sin embargo, conforme se avanzaba en la elaboración del código se llegó a un punto en el que este servidor no cumplía uno de los requisitos principales. Esto se concluyó debido a que los productos a descargar para cada conjunto, denominados assets (activos), en el caso de Sentinel se trataban únicamente de las bandas en las que se componen una imagen, véase [30]. Es decir, con este servidor no se podría disponer de todos los datos originales de un producto (esto es, ficheros XML, informes de controles de calidad, fichero de metadatos, etc).

AMAZON WEB SERVICE (AWS)

Al inicio de este estudio, se clasificó a este repositorio como uno de los mejores a priori. Esta suposición fue debida a que presenta una ventaja que conseguía diferenciarlo del resto de plataformas, y es que al disponer de sus propios algoritmos para la adquisición de imágenes, filtra aquellas imágenes que no pasaban los niveles de calidad en el OLQC y por tanto no se hacían públicas, quedando a disposición de usuario aquellas sin errores.

Sin embargo, estos productos disponibles no mantienen la estructura acordada en la tercera condición, como se puede observar en la Figura 21. Es notable la carencia de archivos si se compara con la representación en la Figura 20.

Aun así, la característica principal de AWS es el almacenamiento en la nube de las imágenes descargadas desde este, en una cuenta personal de usuario, por lo que se contempló esta opción. Utilizando líneas de comandos propias (aws), es posible la descarga del producto (en formato zip) en su totalidad. No obstante esta alternativa requiere la activación de “Requester Pays”, es decir, solicitudes tipo GET, POST, LIST, COPY y PUT que tienen un pequeño costo, lo cual incumple el requerimiento de gratuidad pactado por los socios. [31]

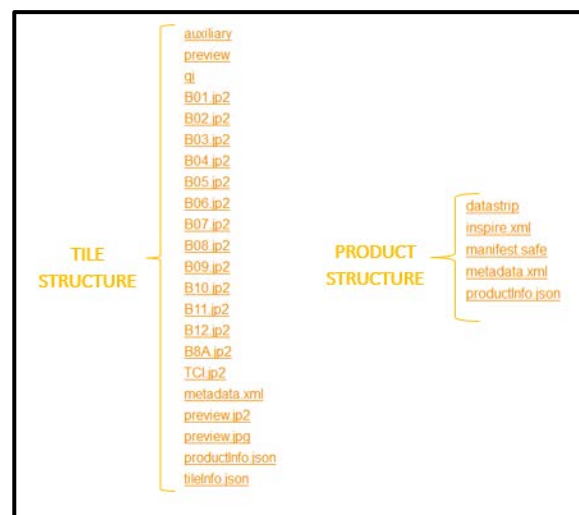


Figura 21: Estructura de contenidos de productos Sentinel 2 en AWS

GOOGLE CLOUD PLATFORM

El problema que presenta este servidor para no convertirse en el proveedor de imágenes para PyrenEOS es muy semejante a uno de los que mostraba el anterior. De nuevo, se trata de un servicio en la nube, donde las imágenes descargadas desde esta plataforma pueden ser almacenadas. Sin embargo, los productos se encuentran en un repositorio de Google que no es de acceso público, sino que es necesario estar en posesión de una clave para acceder a la API. Para obtener esa clave, es de obligado cumplimiento habilitar la facturación en el proyecto, es decir, aunque se trate de una pequeña cantidad, la descarga de imágenes supondría un costo. [32]

EO CLOUD FERRO

Al igual que sus semejantes en cuanto a topología, el acceso a los datos se debe realizar a través de una máquina virtual (VM) creada desde un proyecto establecido previamente en la cuenta de usuario. A pesar de que la creación de la primera VM es gratuita, el espacio ofrecido para el almacenamiento es limitado y esta gratuidad expira en un plazo de 12 meses. Por consiguiente, debido al incumplimiento de la primera condición, no se puede utilizar [33]

4.2.4. Herramientas y Metodología utilizadas

Los restantes servidores (oficiales, mirrors nacionales, ASF y THEIA) necesitan un estudio más específico de acuerdo con distintos parámetros considerados como factores claves a la hora de escoger un proveedor u otro. Con este propósito, se han desarrollado una serie de herramientas flexibles que permitan testar en cualquier tipo de entorno los tiempos de descarga y velocidades para ese escenario

en concreto, además del tiempo de adquisición de una imagen, es decir, en qué momento un repositorio tiene disponible un producto.

La carpeta DOWNLOAD_SERVER, que estará destinada a su instalación en la máquina servidora, contiene otra llamada Web_Repositories_Test, existiendo en ella un conjunto de archivos por cada repositorio a estudiar, como queda ilustrado en la Figura 22.

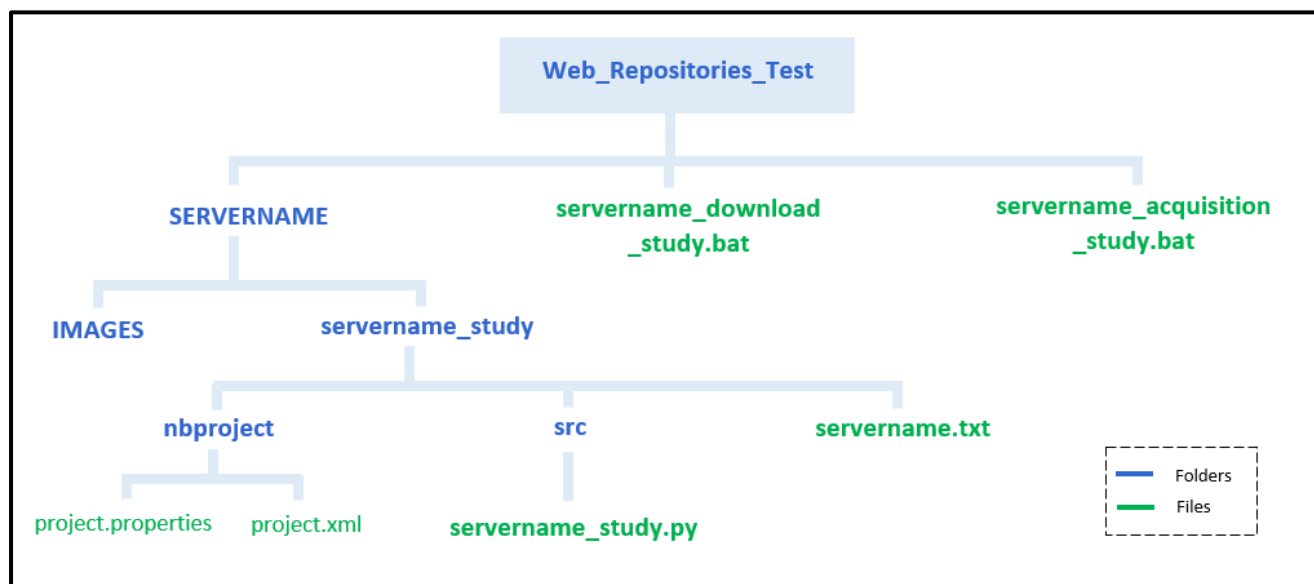


Figura 22: Estructura general de los archivos existentes por cada repositorio dentro de Web_Repositories_Test

A pesar de no estar representada, esta carpeta también contiene una general denominada TIMES, donde se almacenarán los archivos con las soluciones de los estudios de todos los repositorios.

El diseño de los proyectos Python está basado en el código opensource desarrollado por Oliver Hagolle y Álvaro Huarte para la plataforma PEPS [34]. Considerando eficiente la arquitectura de este e intentando respetarla, ha sido necesario adaptar cada script a las necesidades de los repositorios en concreto. A modo de ejemplo, estos ficheros realizan una primera consulta con ciertos parámetros como son plataforma, rango de fechas, etc, obteniendo como respuesta un archivo json, a excepción de CODEDE que devuelve un xml. En él, figuran los enlaces y propiedades de los productos que encajan con dicha consulta, siendo para cada servidor una nomenclatura y organización distinta.

Estos json se almacenan en la carpeta del servidor correspondiente, con diferente nombre entre el obtenido por el estudio de descarga y el de adquisición, a fin de evitar colisiones si se ejecutan al mismo tiempo estos análisis.

Se han añadido a estos más opciones a introducir como argumentos para poder realizar los correspondientes estudios, "if_new" para que se desarrolle el estudio de nuevas adquisiciones de imágenes y "download_time" para que los resultados sean el tiempo de descarga de un producto (escogido por argumentos desde el archivo .bat) pero sin ser almacenado, no ocupando espacio en disco. La programación de estas funcionalidades se corresponde con la Figura 23 en la siguiente página.

Además, dado que la mayoría de los servidores tienen un límite de resultados a devolver (algunos 100 y otros 500), en el caso del análisis de nuevas imágenes, se tiene que cambiar la fecha de inicio de forma dinámica, la forma aparece en la Figura 24, con el fin de no obtener siempre los mismos primeros 100 resultados dados unos rangos concretos.

```

if(options.download_time):
    comparison_file = open("%s\TIMES\Austrianm_download_time.txt"%(options.current_path),"a")
    print "deleting product stored"
    os.remove("%s/%s.zip"%(options.write_dir,prod))
    finish = datetime.datetime.now()
    seconds = int((finish-start).total_seconds())
    comparison_file.write("*****\n")
    comparison_file.write("Product: %s \n Start: %s Finish: %s\n"%(prod,start,finish))
    comparison_file.write("Seconds: %s \n"%(seconds))
    comparison_file.close()
elif (not(file_exists) and options.if_new):
    with open("%s\TIMES\Austrianm_acquisition_time.txt"%(options.current_path), 'r') as infile:
        new_prod = True
        for line in infile:
            prod_attr = line.strip().split(" ")
            prod_name = prod_attr[2]
            if (prod_name == prod):
                new_prod = False
    if(new_prod):
        acquisition_file = open("%s\TIMES\Austrianm_acquisition_time.txt"%(options.current_path),"a")
        current_date=datetime.datetime.now()
        acquisition_file.write("New product: %s ***** Acquisition time: %s\n"%(prod,current_date))
        acquisition_file.close()

```

Figura 23: Código ejemplo en Python para la extracción del tiempo de descarga y la fecha de obtención de un nuevo producto

```

if options.if_new:
    currentdate = datetime.datetime.today()
    start_date=options.start_date
    inputstartdate = datetime.datetime.strptime(start_date, "%Y-%m-%d")
    if(currentdate>inputstartdate):
        difference = int((currentdate-inputstartdate).days)
        mod = round(difference/7)
        if (mod>0):
            days=timedelta(days=4)
            start_date=currentdate-days
            start_date= "%04d-%02d-%02dT00:00:00.000Z"%(start_date.year, start_date.month, start_date.day)

```

Figura 24: Código en Python para cambiar de forma dinámica la fecha de inicio de período de captura de satélite

A fin de poner a funcionar estas herramientas en cualquier escenario se deben cumplir las siguientes condiciones:

- Sistema operativo Windows
- Mantener la organización y nomenclatura de las carpetas y sus contenidos
- Software Python 2.7 [35], curl [36] y aria2 [37]. El primero será necesario para poder ejecutar desde línea de comandos los scripts creados en el lenguaje Python, mientras que curl y aria2 son dos herramientas de línea de comandos que permiten la transferencia de ficheros con distintos protocolos, siendo necesarios para la descarga de datos desde los servidores.

Este último punto es indispensable ya que sin ellos, no se reconocerían los comandos y los programas no funcionarían. Dado que la plataforma está pensada para operar en el lado servidor y uno

de los requisitos de Gobierno de Navarra era la no instalación de ejecutables, se ofrece la carpeta Enviroment_Tools, ubicada en DOWNLOAD_SERVER, que contiene las librerías necesarias para la ejecución de Python, curl y aria2. Las variables de entorno con estas aplicaciones son establecidas en los ficheros .bat. Es por esto, que se insta a mantener la estructura y nomenclatura tal cual se ofrece.

La metodología utilizada es la ejecución automática de cada uno de los ficheros BATCH, que contiene las instrucciones necesarias para hacer correr los estudios, desde el programador de tareas de Windows. Estas aplicaciones serán lanzadas periódicamente, en concreto, cada 15 minutos para los encargados de obtener los tiempos de adquisición y cada 60 minutos para los tiempos de descarga. En el ANEXO A se explica más detalladamente el procedimiento necesario para ello.

A modo resumen, se presenta a continuación una visión general del diagrama de proceso para los distintos análisis por el que se pondrá a prueba cada servidor.

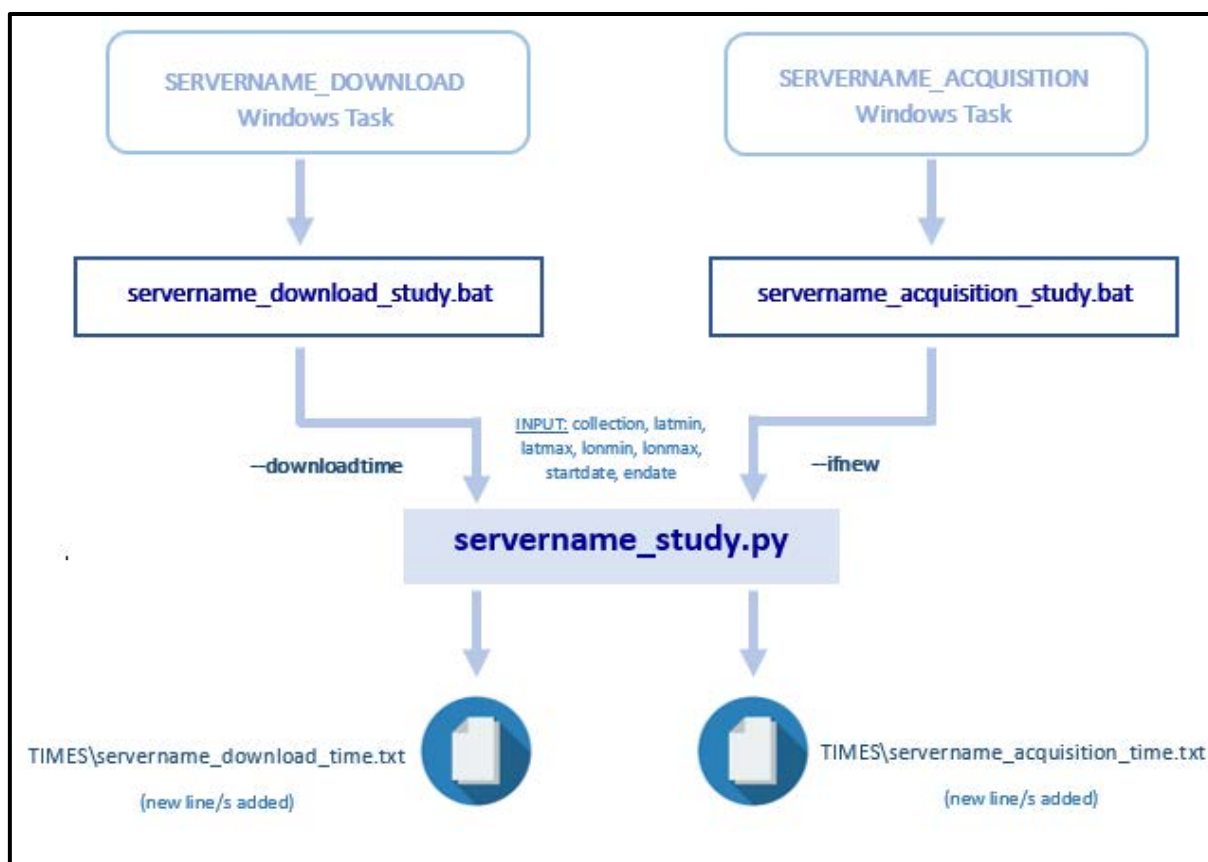


Figura 25: Diagrama de funcionamiento de las herramientas para los estudios de tiempos en cada servidor

4.2.5. Entorno de evaluación

La ejecución de las herramientas explicadas en el apartado anterior se ha realizado en un entorno domiciliario no controlado con un ordenador de mesa personal, conectado continuamente a la corriente. Las características de interés son las siguientes:

- Sistema operativo Windows 10 Enterprise Profesional
- Equipo basado en x64 ACPI
- Conexión Ethernet

- Velocidad contratada asimétrica, 30 Mbps de subida y 300 Mbps de bajada
- CPU dual-core, 3.7 GHz
- Memoria RAM instalada de 8 GB
- Disco rígido IDE con única partición (Unidad C:/) con capacidad de 223,1 GB, quedando libres 145,7 GB.

Cabe mencionar el hecho que durante el período de pruebas no se ha ejecutado ninguna descarga o petición que pudiera afectar al límite de ancho de banda disponible.

4.2.6. Comparativas

Las herramientas del apartado 4.2.4 se mantuvieron en funcionamiento, recogiendo datos, desde el 20 de Abril de 2017 hasta el 31 de Mayo del mismo año. Para los estudios de adquisición se seleccionó todo el marco de Navarra y País Vasco, con el fin de tener suficientes datos para realizar las estadísticas. En el caso del estudio de descarga, se escogieron parámetros más ajustados a fin de bajar un solo y mismo producto y no tener corriendo el script más tiempo del necesario.

Además de las comparativas que con estos datos se podrán obtener, se realizará una primera sobre los productos y niveles Sentinel que cada repositorio web ofrece. Esto es debido a que el fin de estudio será realizar un árbol de decisiones por cada producto, es decir por cada misión Sentinel, poniendo en primer lugar aquellos que mejores resultados obtengan.

4.2.6.1. Productos Sentinel disponibles

Inicialmente se presenta la Tabla 8, aportando una visión general de los productos disponibles en cada uno de los servidores encontrados según la colección, es decir, según la misión de la que provienen dichas imágenes. Este punto será importante para conocer desde dónde se descargarán unas imágenes u otras.

	AUSTRIANM	ASF	AWS	CEDA	CODA	CODE-DE	COMMON N.M	EO CLOUD	GOOGLE	HNSDMS	PLANET	PEPS	SCIHUB	SEDAS	SENTINEL-HUB	SWEA	THEIA	USGS
SENTINEL 1A																		
SENTINEL 1B																		
SENTINEL 2A																		
SENTINEL 2B	Lanzamiento 7 de Marzo de 2017																	
SENTINEL 3A Land																		
SENTINEL 3A Ocean																		

Tabla 8: Tipos de imágenes disponibles según la misión satélite en los distintos servidores

Es notable el hecho que el 90% de los repositorios tienen los conjuntos de datos correspondientes al satélite Sentinel 2A, y seguramente se dará la misma situación una vez que se reciban las primeras imágenes oficiales del 2B. Este gran porcentaje cobra sentido ya que las capturas del Sentinel 2, de alta resolución espectral y espacial, son las más utilizadas debido a dedicarse al monitoreo de la cubierta terrestre.

Para finalizar este apartado, se diferenciarán los niveles de tratamiento de las imágenes que cada punto de acceso ofrece, para hacerlo más útil, sólo de los no descartados en el punto 4.2.3. Esta distinción será un punto considerable ya que se puede dar la situación que no se requieran de absolutamente todos los niveles de productos, si no que será de interés contar con los más tratados o los que estén más en bruto.

	AUSTRIANM	ASF	CODA	CODE-DE	HNSDMS	PEPS	SCIHUB	THEIA
SENTINEL 1A	L0_RAW L1_GRDH	L0_RAW L1_GRDH		L1_GRDH L1_SLC L2_OCN	L0_RAW L1_GRDH L1_SLC L2_OCN	L1_GRDH L1_SLC L2_OCN	L0_RAW L1_GRDH L1_SLC L2_OCN	
SENTINEL 1B	L1_SLC L2_OCN	L1_SLC L2_OCN						
SENTINEL 2A	L1C			L1C	L1C	L1C	L1C L2A	L2A
SENTINEL 3A Land	L1_SAR L2_SAR OLCI SLSTR					L2_SAR OLCI SLSTR	L1_SAR L2_SAR OLCI SLSTR	
SENTINEL 3A Ocean			L1_SAR L2_SAR OLCI SLSTR					

Tabla 9: Tipos de imágenes disponibles según el nivel de producto en los servidores a estudiar

A pesar de ser cierto que SCIHUB cuenta con las imágenes del Sentinel 3A, esta colección se encuentra actualmente bajo credenciales de invitado, en situación pre-operacional, y no son accesibles de momento a través de un script.

4.2.6.2. Tiempos de adquisición

En este apartado se procederá a analizar los resultados obtenidos por las herramientas correspondientes al estudio de adquisición. El objetivo es comprobar la rapidez con la que los distintos repositorios obtienen las imágenes y las dejan accesibles al usuario con el fin de disponer de estas lo más rápido posible en el repositorio propio de Gobierno de Navarra. Cuanto mayor sea este tiempo, más imágenes se acumularán para descargar y por tanto el período de tiempo en el que la máquina de descarga está ocupada en un día es mayor.

Debido a la diferencia de colecciones que cada uno de estos posee, se realizará una gráfica comparativa por cada misión Sentinel sirviendo como referencia, tiempo 0, el momento en que el satélite captura la imagen.

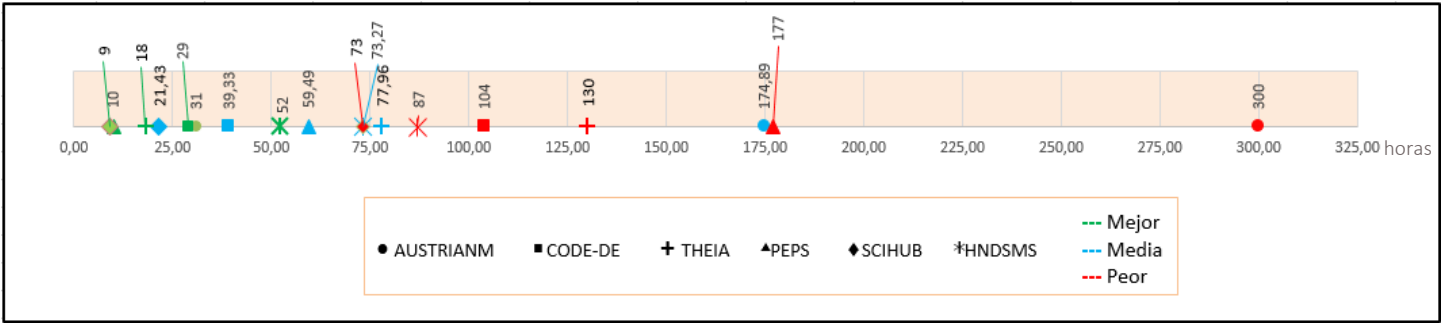


Gráfica 1: Eje cronológico de adquisición de productos SENTINEL 1 de los servidores web

En la representación superior son notables los resultados obtenidos por el repositorio oficial, SCIHUB, siendo los tres tipos de estadísticas (mejor, media y peor) muy bajas y parejas entre sí. Sin embargo, el propósito de este estudio no es encontrar una única solución.

Lo interesante de estos datos es que la mayoría de los repositorios restantes se mantienen en un mismo rango de tiempo. Ejemplos destacables son las grandes diferencias de peores resultados que obtienen PEPS y AUSTRIANM y los registros de HNSDMS y ASF. El primero de ellos, a pesar de ser el perteneciente a la peor media, el resto entre esta y su mejor y peor marca es de las más pequeñas en comparación, es decir, adquiere imágenes en tiempos constantes. Referente al segundo nombrado, ASF, destaca, además de los bajos resultados, el hecho, encontrado en el estudio, de poner a disposición las imágenes nuevas por primera vez a unas horas marcadas del día, siendo estas de 8:00 a 8:30 y de 22:30 a 00:00.

Estas deducciones son respecto al comportamiento de los servidores con los datos del Sentinel 1. Es por esto que se tendrá que seguir con las gráficas comparativas con el fin de comprobar si estos patrones se mantienen en el resto de conjuntos de datos.



Gráfica 2: Eje cronológico de adquisición de productos SENTINEL 2 de los servidores web

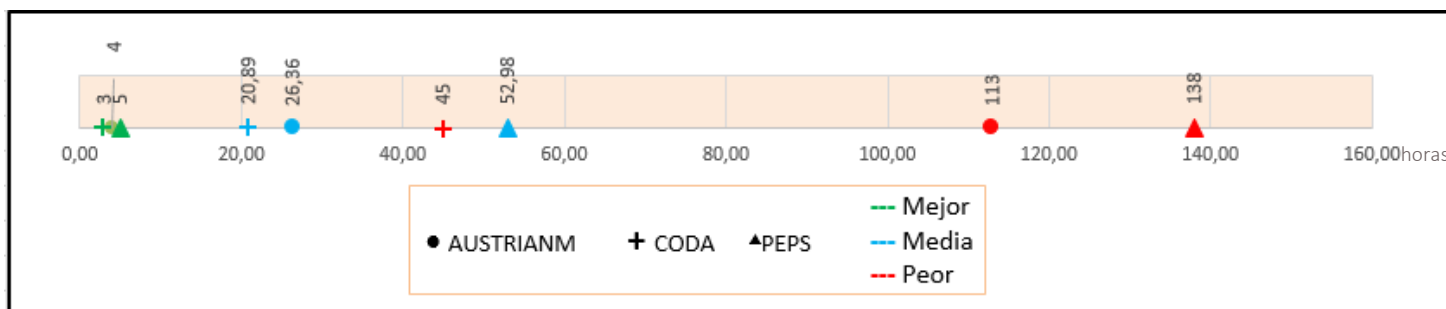
La gráfica dada muestra un incremento general de los resultados anteriores. La excepción a este hecho lo crea HNSDMS, que obtiene iguales tiempo, siendo más contundente la hipótesis de su constancia en la hora programada para la adquisición de las imágenes.



AUSTRIANM es el servidor que presenta mayor aumento, teniendo un retardo medio respecto del anterior de 100 horas, es decir, aproximadamente 4 días. Punto desfavorable teniendo en cuenta la resolución espectral de los satélites, recordando que para el caso del Sentinel 2 era de 5 días. Además de esta posible desventaja, se ha deducido de los registros obtenidos por los estudios, la limitación de imágenes que posee. La tipología de este servidor es National Mirror, que como ya se ha explicado, puede contener colecciones parciales según la zona geográfica, por lo que al ser el área de estudio un punto alejado respecto Austria, la adquisición de estas, demora más tiempo al no ser fundamentales. El motivo de no apreciar esta limitación en la Gráfica 1 es debida a que las imágenes del Sentinel-1 y 3 abarcan mayores áreas que las del Sentinel 2, de tal forma que para estas últimas una imagen que comprenda Francia nunca capturará alguna zona de España. Situación semejante ocurre con la plataforma CODE-DE.

Otro punto crítico del servidor austriaco con respecto a su capacidad es el hecho que no solo tiene limitación geográfica, sino temporal, es decir, a través de script solo se pueden descargar imágenes de máximo 2 meses de antigüedad. Esto, salvo si se requiere de una pre-carga inicial, no sería crítica en la descarga automática deseada.

Como último comentario, dejando de lado los resultados positivos de los servidores, cabe destacar las marcas obtenidas por THEIA teniendo en cuenta que sus imágenes ofrecidas no son las capturas por el satélite (o recibidas por SCIHUB) sino que además sufren un pre-procesado que incrementa el tiempo de adquisición del producto final.



Gráfica 3: Eje cronológico de adquisición de productos SENTINEL 3 de los servidores web

Curiosamente, en la Gráfica 3, es el repositorio AUSTRIANM el que obtiene las mejores marcas, debido al rango abarcado por las imágenes y reforzando la conclusión anteriormente mencionada. No se deben tener en cuenta comparaciones entre CODA y este, ya que los productos son de diferente naturaleza.

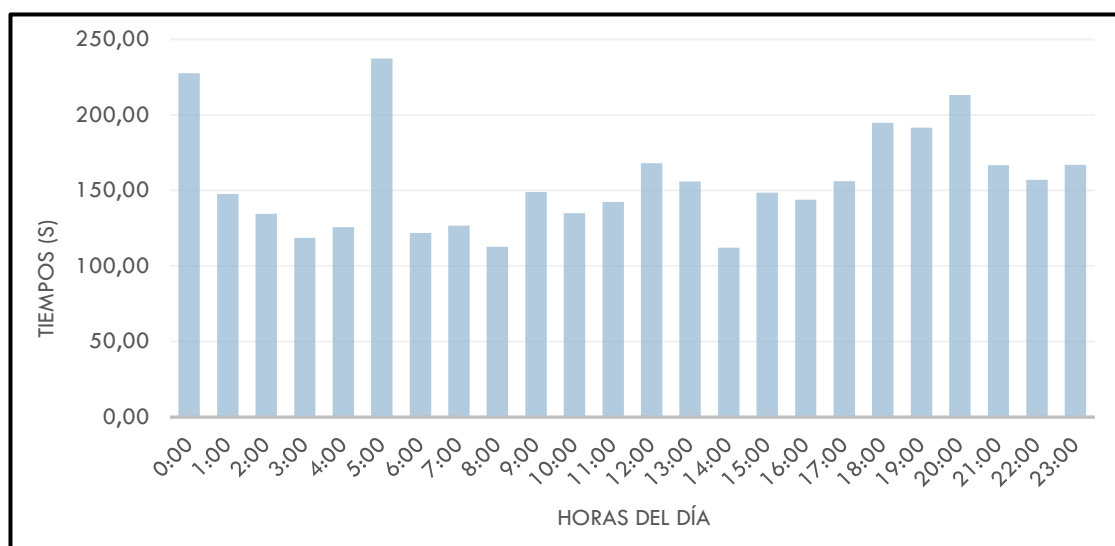
Las deducciones obtenidas no serán suficientes para jerarquizar los servidores del estudio en cada caso. Para ello, habrá que apoyarse en los resultados del siguiente apartado.

4.2.6.3. Tiempos de descarga

Esta sección presenta los análisis realizados a partir de los datos obtenidos por parte de las herramientas correspondientes a los estudios de descarga. El objetivo, en este caso, es comprobar la rapidez o lentitud en la descarga de las imágenes desde el servidor al directorio del usuario, deduciendo por tanto el tiempo invertido en esta.

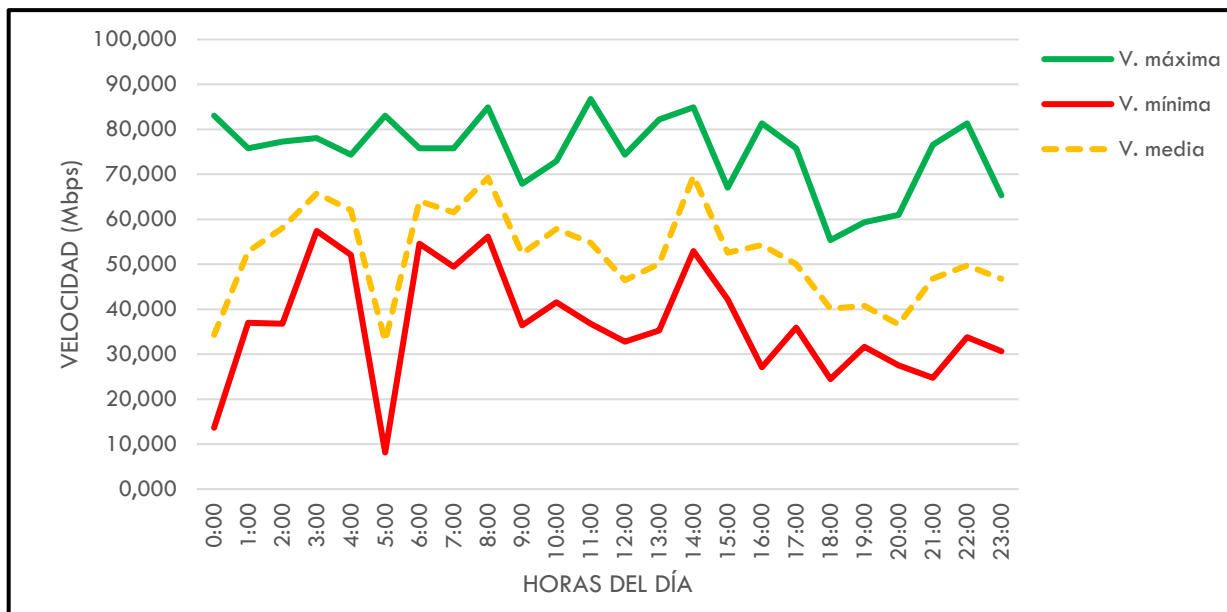
Para ello, se elaborarán una serie de gráficas por cada servidor, contrastando los mejores y peores resultados, y otras de carácter comparativo global. La utilidad de estas gráficas será la deducción del comportamiento de cada servidor según las horas del día y la limitación en la conexión a lo largo del día, concluyendo si es la velocidad contratada en el entorno o la velocidad de red o del servidor la que reduce los tiempos resultantes.

ASF – (ALASKA SATELLITE FACILITY)



Gráfica 4: ASF - Media estadística de los tiempos de descarga a lo largo del día

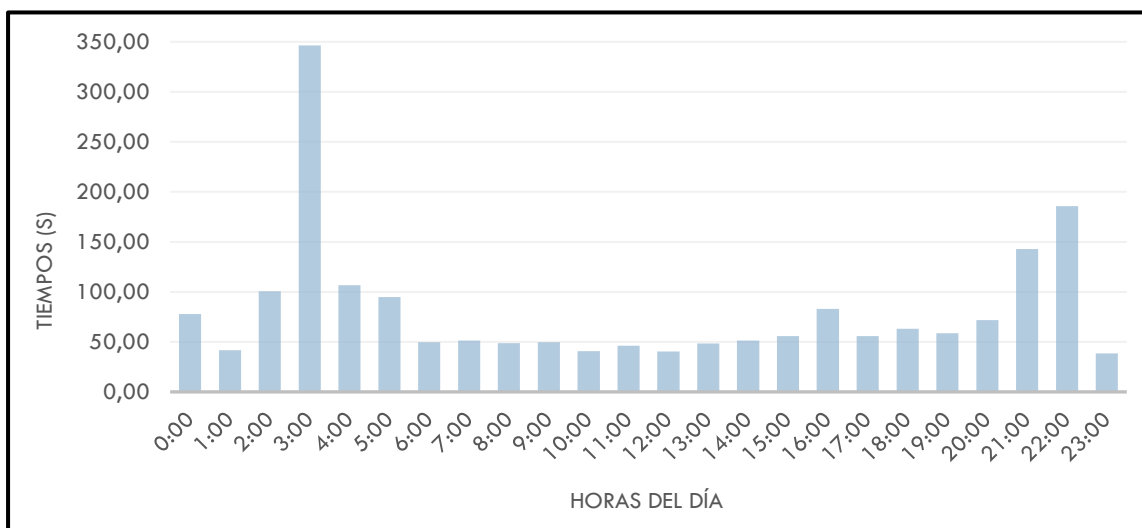
El gráfico superior muestra el tiempo invertido en la descarga, siendo el dato de cada hora la media de esa misma hora del día en los siete de la semana para un producto de 976 MB. Con estos resultados, se pueden deducir las mejores horas para bajar imágenes en este repositorio (entre la 1:00 y las 11:00, descartando las 5:00). Sin embargo la hora final a la que se programará la descarga debe ser aquella en la que se obtengan adecuados tiempos para prácticamente la mayoría de los repositorios



Gráfica 5: ASF- Velocidades de descarga a lo largo del día

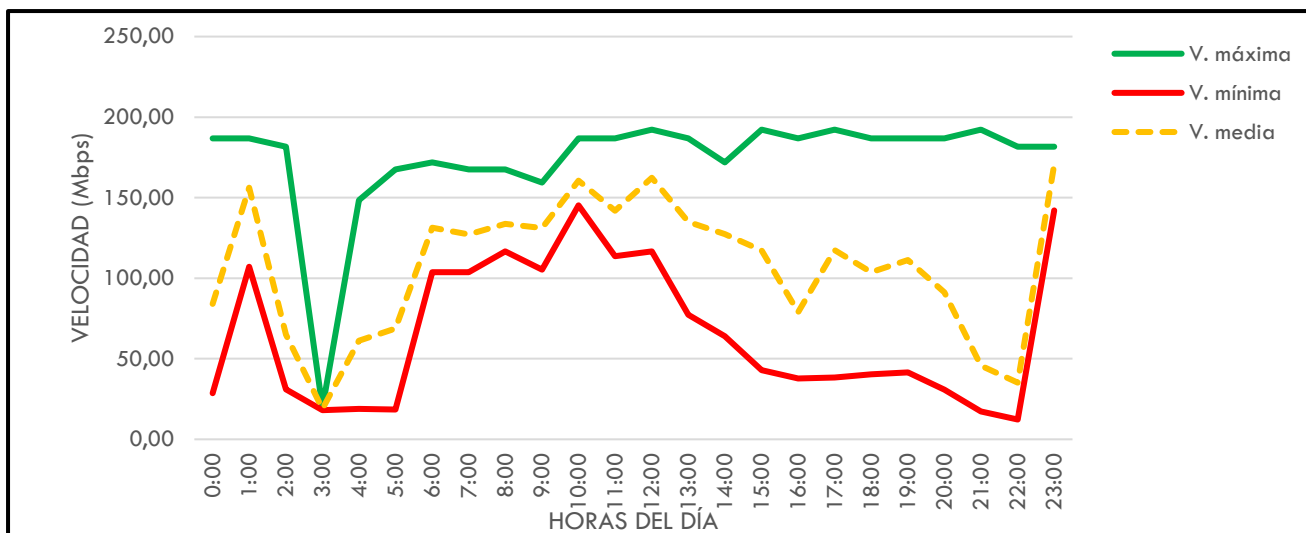
Los resultados de la representación de velocidades en la Gráfica 5, dejan claro que el cuello de botella en la conexión se encuentra en la red o servidor y no en el lado del usuario, ya que el pico de velocidad máxima está lejos de alcanzar los 300Mbps de bajada contratada. Por otro lado, es notable la irregularidad en el comportamiento del servidor, siendo más propenso a tiempos bajos debido a la proximidad entre las velocidades media y mínima obtenidas.

AUSTRIA NATIONAL MIRROR



Gráfica 6: AUSTRIANM - Media estadística de los tiempos de descarga a lo largo del día

La plataforma austriaca, pese a los resultados del apartado 4.2.6.2, obtiene unas marcas de tiempos en general relativamente bajas para una imagen de 817MB. Exceptuando el apreciable pico a las 3:00, con gran diferencia respecto al resto de datos. En la Gráfica 7 es visible el motivo de este máximo.



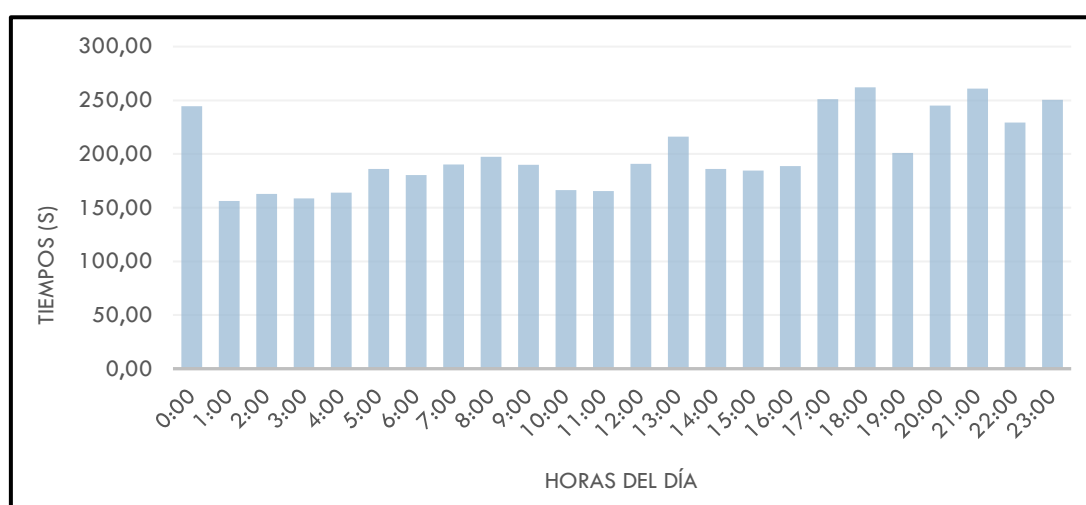
Gráfica 7: AUSTRIANM - Velocidades de descarga a lo largo del día

Visiblemente, en la representación queda justificado el máximo obtenido en la Gráfica 6, coincidiendo las velocidades máxima, media y mínimas en ese punto. Este hecho hace plantear la hipótesis que en torno a las 3:00 el repositorio baja el ancho de banda de salida destinado a la descarga de imágenes debido a posibles labores de mantenimiento, servicio de copia de backup, servicio de descarga con preferencias, etc.

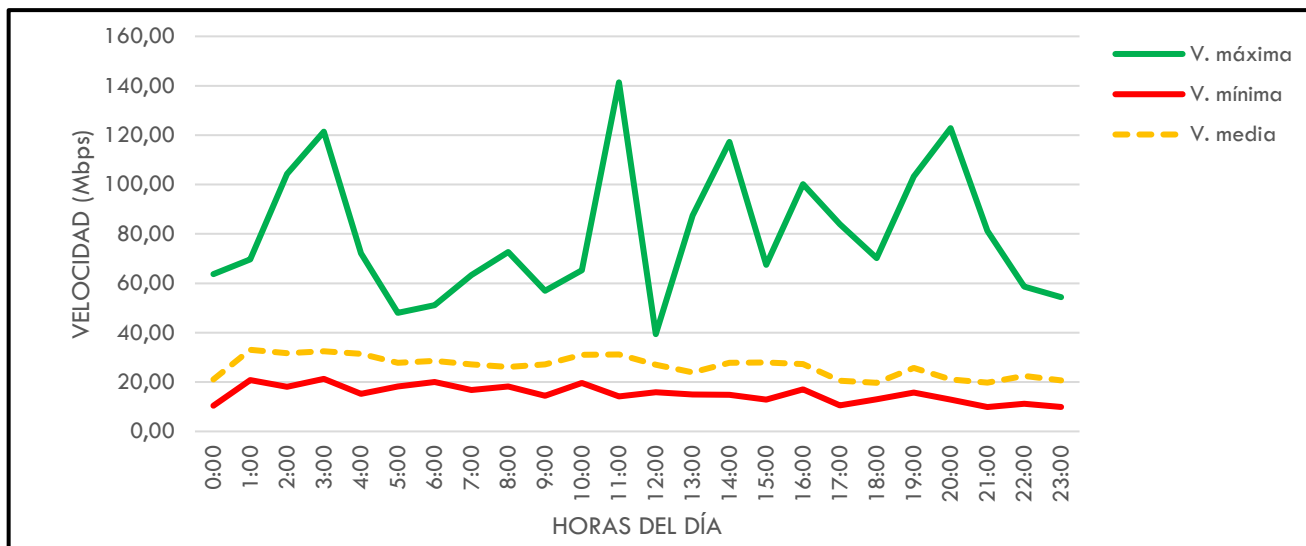
Destaca igualmente la constancia en la velocidad máxima obtenida en el 90 % de las horas, de nuevo sin alcanzar los 300 Mbps pero obteniendo resultados satisfactorios. Esto se traduce en que la rapidez de descarga, siempre y cuando se contrate una velocidad superior a los 200Mbps, será independiente de la velocidad de bajada del usuario.

Por último, cabe mencionar las marcas obtenidas entre las 6:00 y las 12:00, siendo positivos debido a la estrecha diferencia entre las velocidades máxima y mínima, lo cual deriva en un comportamiento y rendimiento del servidor constante.

CODA – (COPERNICUS ONLINE DATA ACCESS)



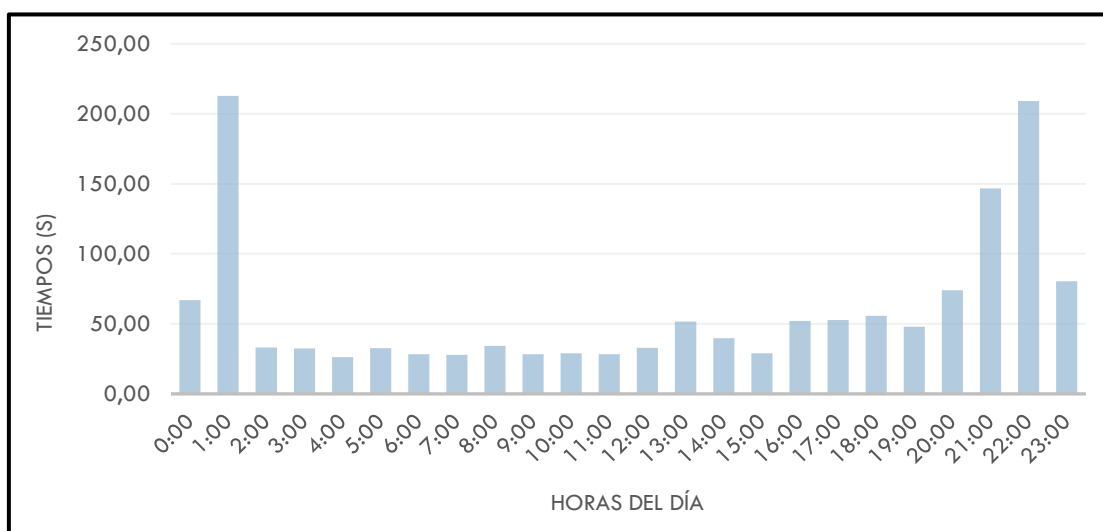
Gráfica 8: CODA- Media estadística de los tiempos de descarga a lo largo del día



Gráfica 9: CODA - Velocidades de descarga a lo largo del día

Extrayendo información de la Gráfica 8, cabe señalar la tendencia a tiempos relativamente altos, del orden de 3 minutos, manteniéndose en esa línea prácticamente todas las horas. Esta tendencia se puede explicar con la representación de las velocidades de este servidor en la Gráfica 9. Como se puede observar, las trazas correspondientes a la velocidad mínima y media son muy similares, existiendo poca distancia entre estas. Comparando con la respectiva a la velocidad máxima, la similitud anterior se traduce en la poca probabilidad de obtener los resultados más altos en cuanto a velocidades se refiere.

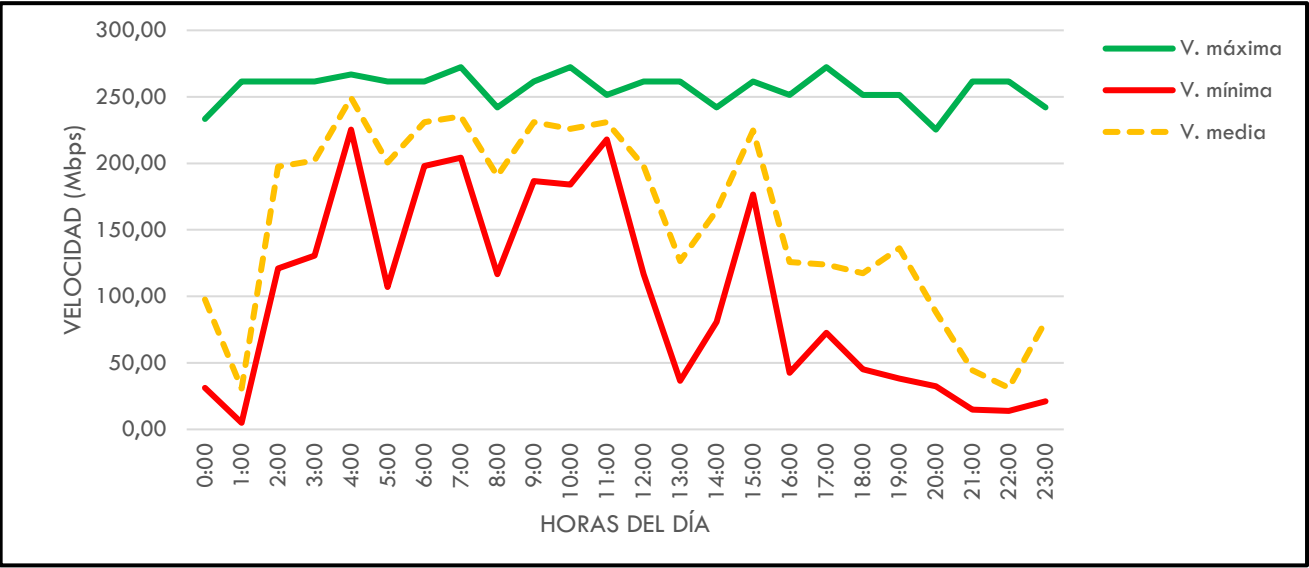
CODE-DE – (COPERNICUS IN DEUTSCHLAND)



Gráfica 10: CODE-DE - Media estadística de los tiempos de descarga a lo largo del día

Similar apunte que el realizado en este apartado para el caso de AUSTRIANM, se hace notable los bajos resultados obtenidos, situación favorable, a excepción del rango de horas comprendidas entre

las 21:00 y 1:00 donde el tiempo invertido en la descarga de un producto de 817 MB incrementa en aproximadamente un 400% respecto la mayoría de los datos.

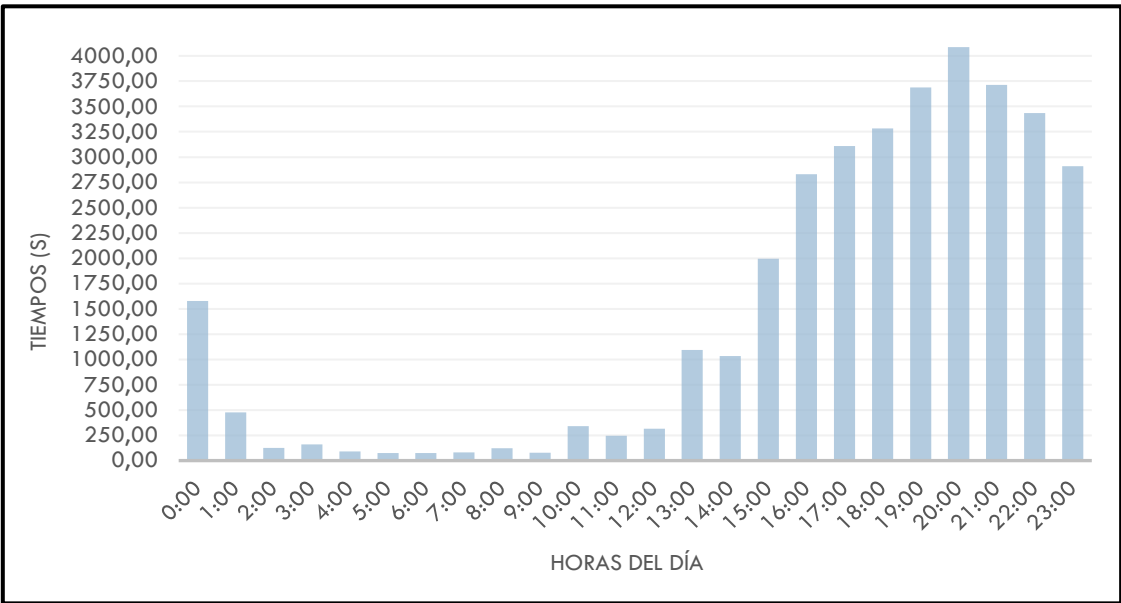


Gráfica 11: CODE-DE - Velocidades de descarga a lo largo del día

Al igual que se ha dado en otro caso anterior, destacan las velocidades máximas obtenidas a lo largo del día, manteniéndose constante en torno a 250Mbps, aproximándose a la velocidad de bajada en el lado del usuario en el caso del entorno de evaluación dado.

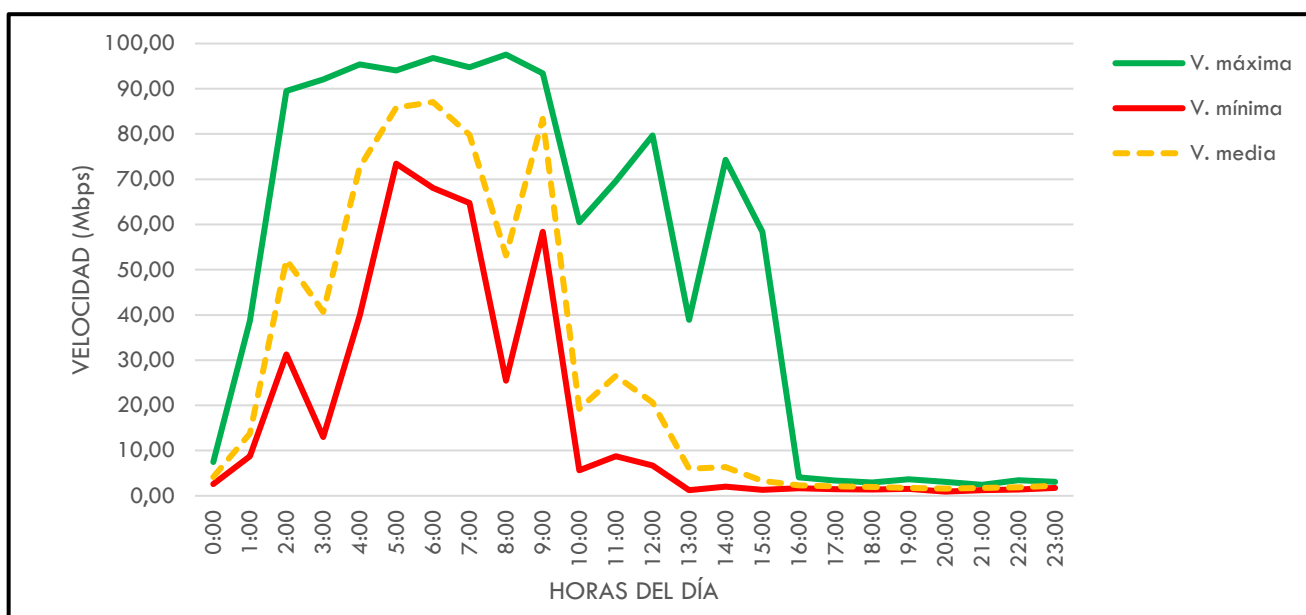
En última instancia, señalar la diferencia de valores de las peores velocidades entre las horas de 4:00 a 12:00 y las incluidas entre las 17:00 y la 1:00, teniendo este último periodo de tiempo mayor probabilidad de alto tiempo invertido en la descarga. En el rango de la madrugada, es relevante la pequeña diferencia entre las mejores y peores marcas, siendo resultados positivos para el estudio.

HNSDMS – (HELLENIC NATIONAL SENTINEL MIRROR SITE)



Gráfica 12: HNSDMS - Media estadística de los tiempos de descarga a lo largo del día



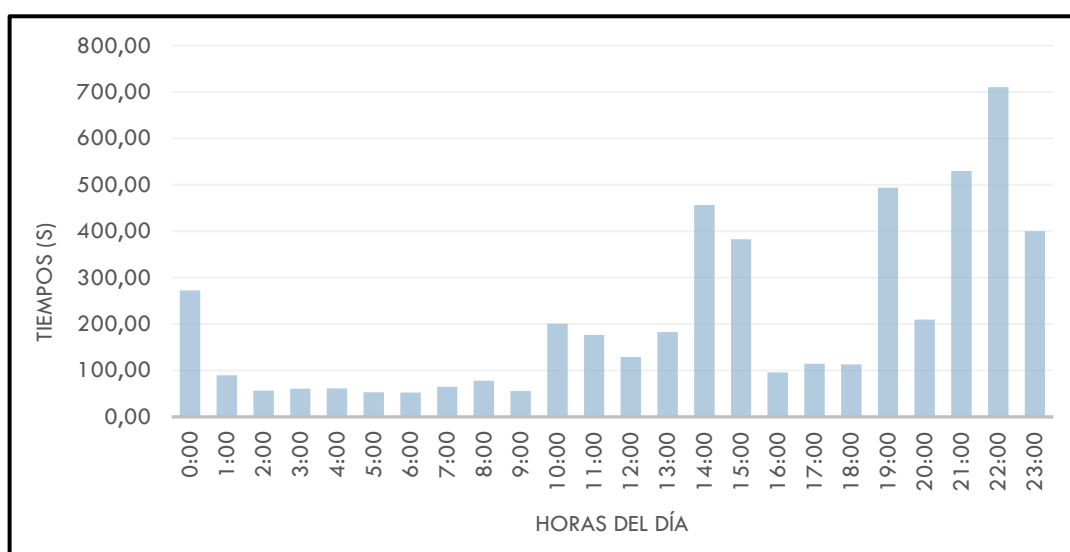


Gráfica 13: HNSDMS - Velocidades de descarga a lo largo del día

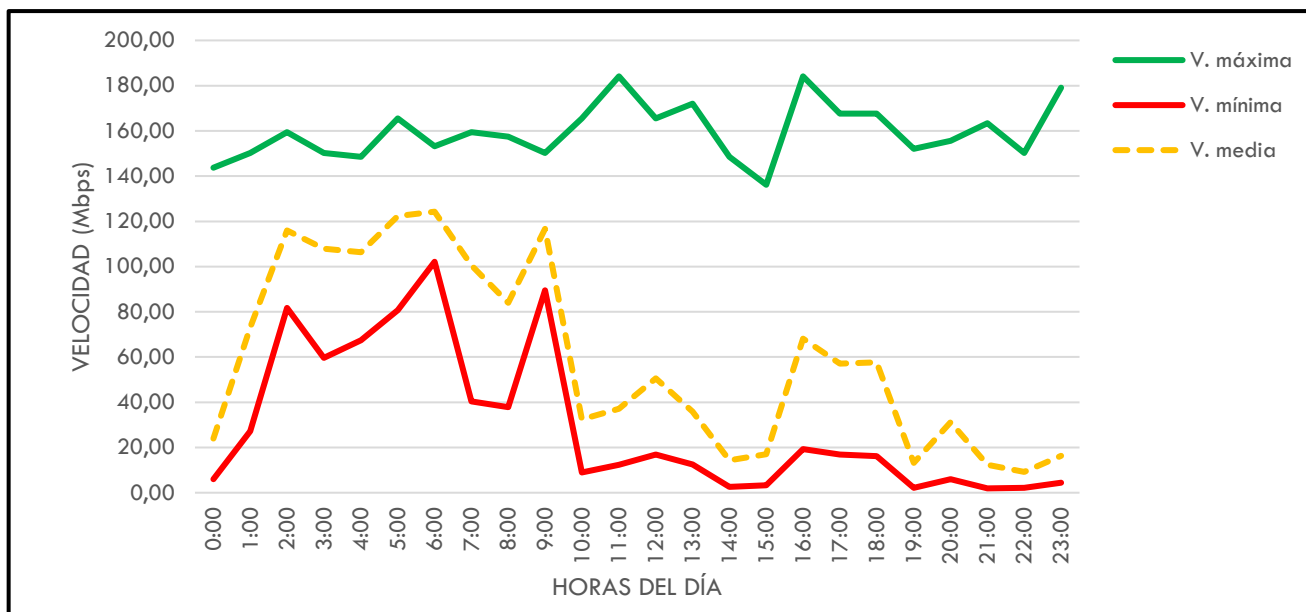
En el caso de la plataforma helénica, existe una muy apreciable diferencia entre el proceso de descarga de una imagen de 817 MB a la tarde y el realizado a la madrugada y mañana, obteniendo unos tiempos de hasta 1 hora 10 minutos, alcanzando velocidades de descarga de 2Mbps. Queda claro por lo tanto que ante los negativos resultados, se evitará la programación de la descarga en este rango de horas.

Aun así, el periodo de tiempo no descartado no destaca tanto por los tiempos obtenidos en comparación otros servidores. No hay que orientarse por la apariencia de la Gráfica 12, ya que los valores mínimos del eje Y son de entre 100 y 250 segundos. Apoyándose sobre la Gráfica 13, queda más claro la velocidad conseguida en el mejor rango de este repositorio, siendo entre 60 y 100Mbps.

PEPS – (PLATEFORME D'EXPLOITATION DES PRODUITS SENTINEL)



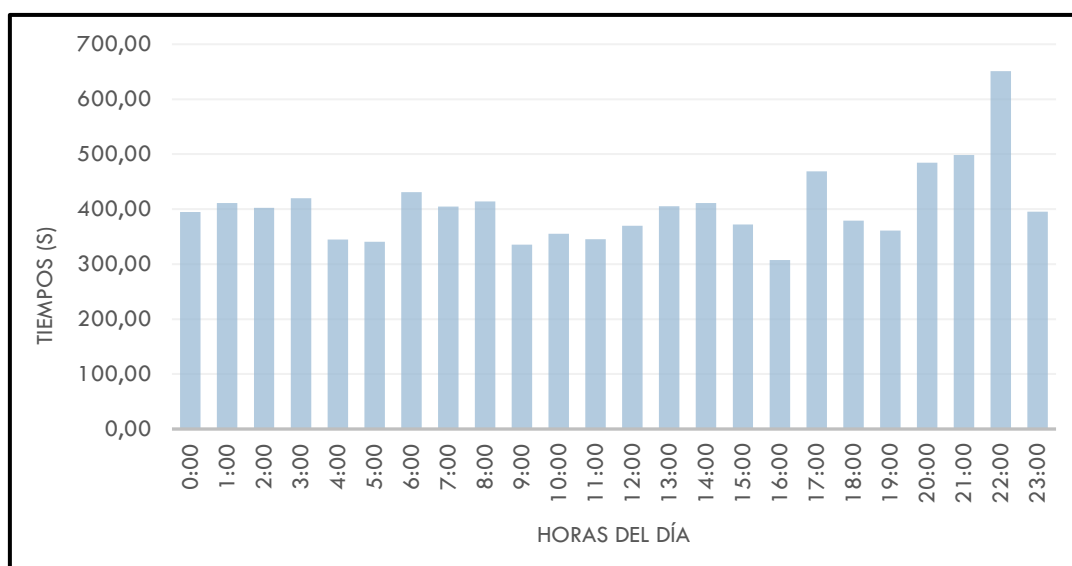
Gráfica 14: PEPS – Media estadística de los tiempos de descarga a lo largo del día



Gráfica 15: PEPS - Velocidades de descarga a lo largo del día

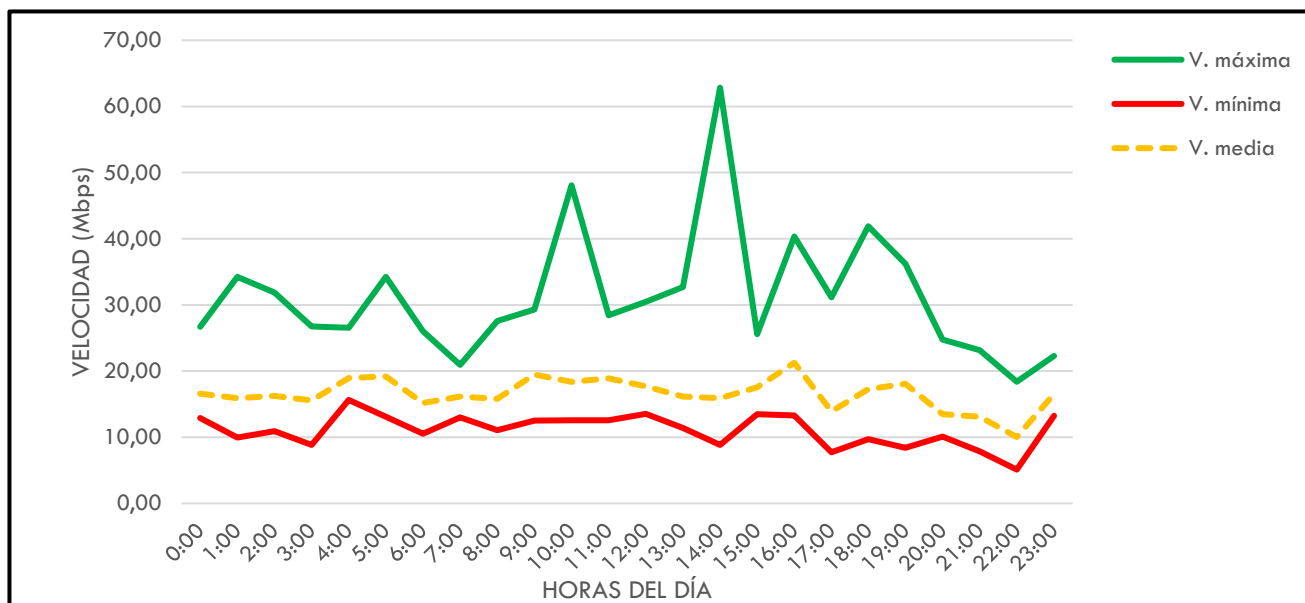
Al igual que anteriores servidores estudiados en esta sección, PEPS cuenta con un periodo de horas en el que la media estadística es prácticamente el doble mejor que el resto de valores. Este rango se corresponde con las horas comprendidas entre la 1:00 y las 9:00, siendo puntos donde la distancia entre las marcas máximas y mínimas es la más pequeña de la gráfica, pero aun así considerable. Este último apunte puede significar que el servidor en cuestión no se comporta de una manera clara, en el rango de tiempos mencionado, existiendo prácticamente iguales probabilidades de obtener una alta o baja marca.

SCIHUB – (SCIENTIFIC DATA HUB)



Gráfica 16: SCIHUB - Media estadística de los tiempos de descarga a lo largo del día

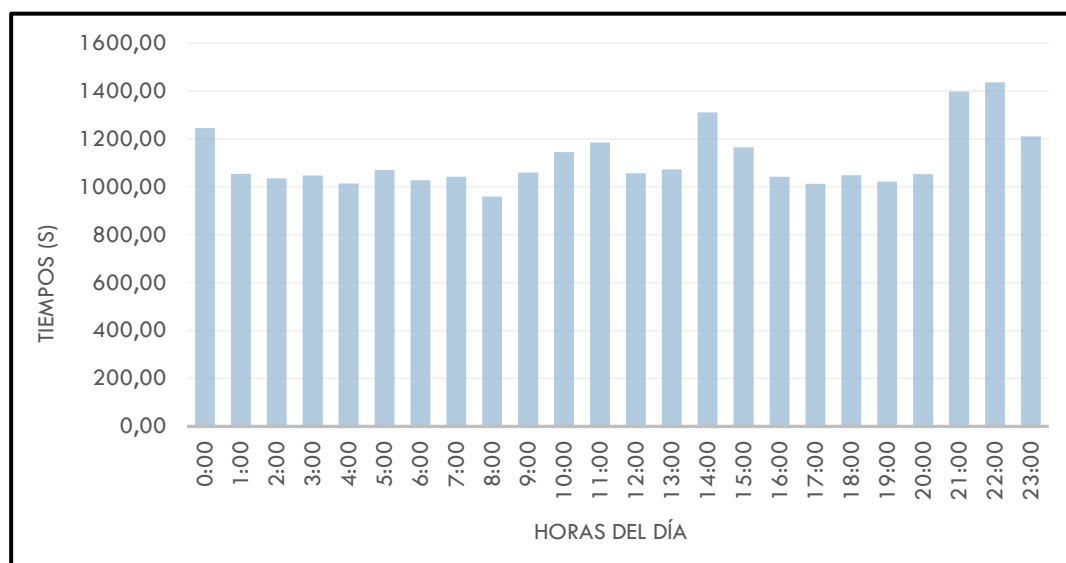
El punto de acceso oficial de la ESA presenta unos altos resultados para la descarga de un producto de 817 MB, siendo notable el parecido con su homólogo de EUMETSAT, CODA. Sin embargo, a diferencia de este, SCIHUB mantiene la línea de tendencia en torno a los 6 minutos por imagen.



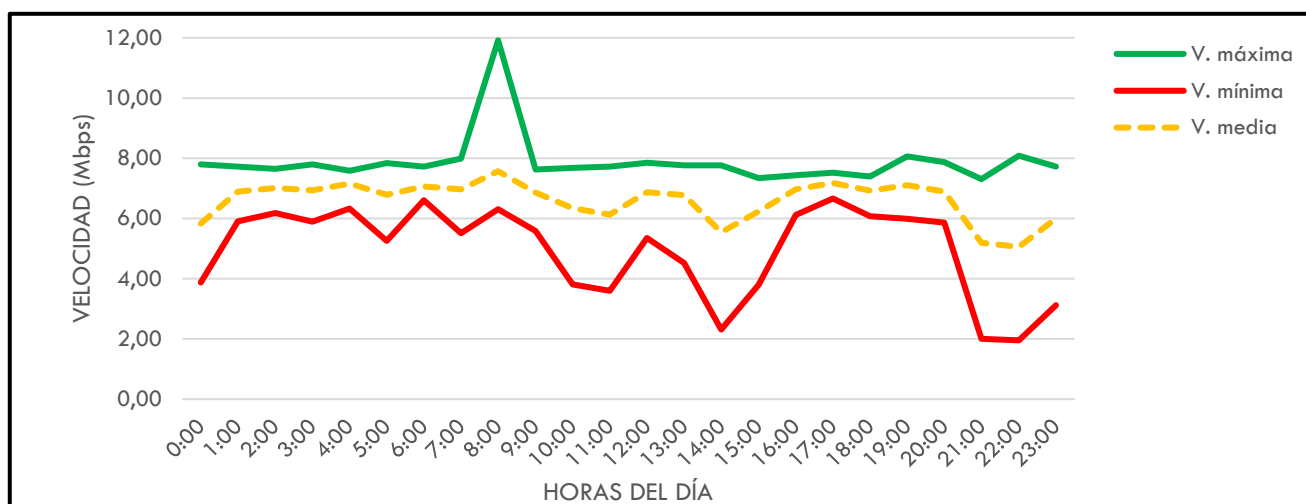
Gráfica 17: SCIHUB – Velocidades de descarga a lo largo del día

La similitud entre las series de velocidad máxima y mínima, hace llegar a la conclusión del rendimiento general siendo menos probable la obtención de los resultados más altos, tendiendo a valores bajos del orden de 20 Mbps.

THEIA



Gráfica 18: THEIA - Media estadística de los tiempos de descarga a lo largo del día



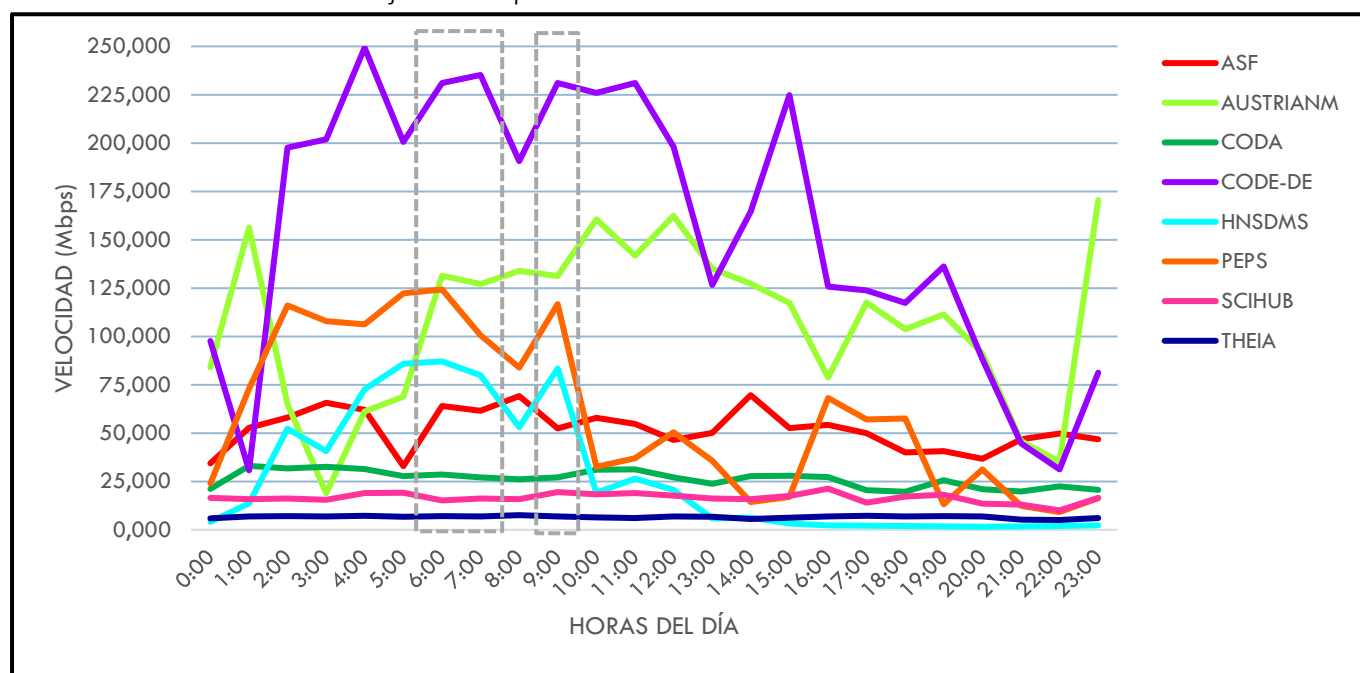
Gráfica 19: THEIA - Velocidades de descarga a lo largo del día

Finalizando con el análisis individual de cada servidor para los tiempos de descarga, se procederá a extraer la información más relevante del repositorio THEIA. Es de obligada mención los altos resultados obtenidos para un producto de 908 MB en la Gráfica 18. Sin embargo, a pesar de esto, tiene el punto favorable de mantenerse constante al igual que se ha visto con otros servidores.

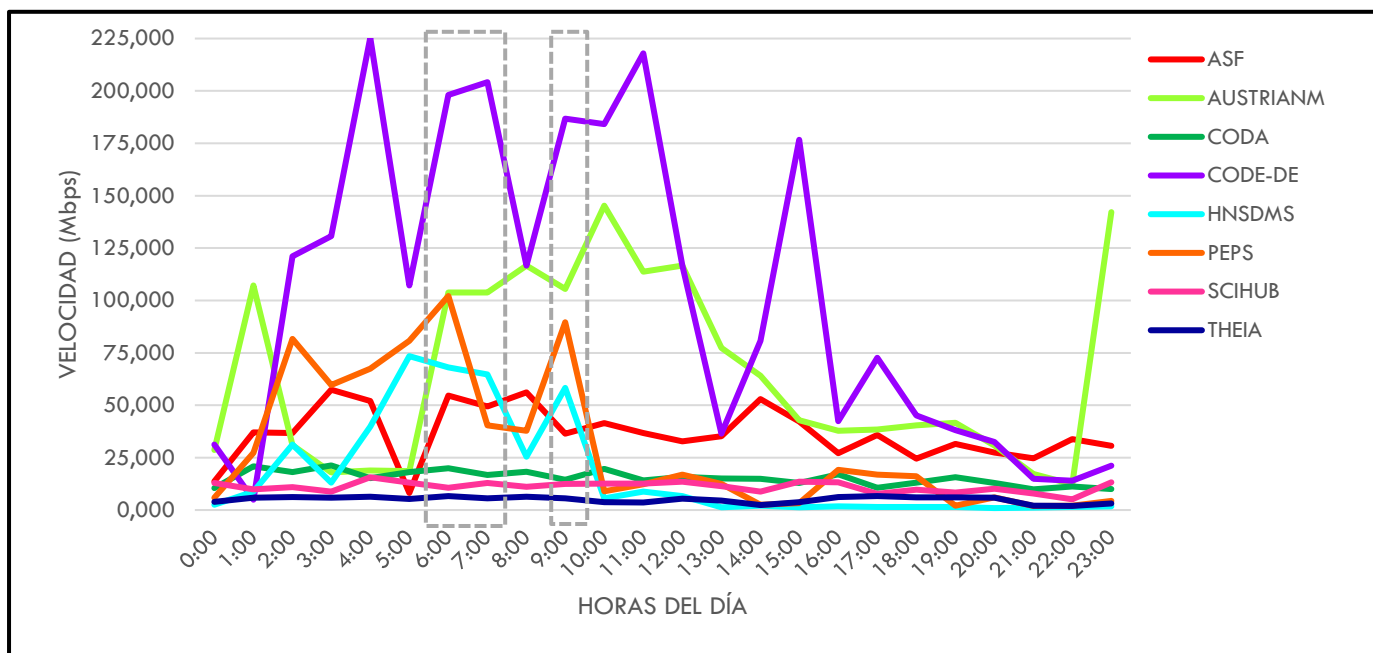
A diferencia del resto, el rendimiento de THEIA es muy similar a lo largo del día, siendo parejas las trazas de velocidades máxima, media y mínima, lo cual asegura conocer previo a la automatización de la descarga, el tiempo que costará la bajada de un producto utilizando este repositorio.

COMPARATIVA GENERAL

Debido a la gran cantidad de información otorgada en este apartado, se presentan a continuación gráficas comparativas de las velocidades media y baja de todos los servidores estudiados con el fin de dotar de una mejor visión para obtener la hora del día más eficaz.




Gráfica 20: Comparativa de velocidades medias de descarga de los distintos servidores



Gráfica 21: Comparativa de velocidades mínimas de descarga de los distintos servidores

Antes de exponer las conclusiones, cabe mencionar la similitud, en las dos gráficas anteriores, entre los dos repositorios oficiales SCIHUB y CODA y la superioridad en velocidad alcanzada de los restantes servidores respecto a estos dos. Con este breve apunte, se procederá a determinar la mejor hora para lanzar el software de descarga de la plataforma PyrenEOS.

Como se puede apreciar, se han añadido dos áreas  en la Gráfica 20 que resaltan aquellas horas cuyas velocidades obtenidas para los distintos repositorios son de las más altas de la gráfica. Sin embargo, no bastará con apoyarse sobre la media. Un buen criterio es situarse en el caso peor, es decir en las velocidades mínimas, escogiendo el rango de tiempo con los mejores valores dentro de ese peor caso. Estas horas con los mejores resultados quedan señaladas nuevamente mediante el recuadro grisáceo en la Gráfica 21, coincidiendo con las obtenidas en la representación anterior: las 6:00, las 7:00 y las 9:00.

Comparando ambas velocidades para las 3 horas, se escoge las 6:00 como hora de arranque del proceso de descarga en el servidor propio, ya que consta de mejores datos, aunque leves, que los correspondientes con las 7:00. A pesar de adquirir velocidades similares con las 9:00, se descarta esta hora a fin de no ocupar el servidor en horario laboral donde es más que seguro que tendrá que requerir capacidad para otros servicios.

4.2.7. Conclusiones de repositorios web

Con las comparativas anteriores, se puede elegir un servidor que sea el proveedor oficial. Sin embargo, la mayoría posee intervalos de mantenimiento, además de los más que típicos fallos de

servidor. Por eso se piensa desarrollar una arquitectura multiservidor en el que cada colección diferente se registrará por un árbol de decisión.

La construcción de este último dependerá de los parámetros del apartado anterior. En este punto se debe tomar una decisión evaluando qué cuestión es más crítica: invertir más tiempo por hora en la descarga de una imagen, tener las imágenes a descargar cuanto antes sin importar la velocidad del repositorio, acumular un total de imágenes y descargarlas de golpe, pero de manera rápida, etc.

La conclusión dependerá de la situación en concreto, siendo en este caso importante el tener las imágenes casi en tiempo real y de la manera más rápida para no tener ocupado al servidor más tiempo del necesario. Sin embargo, curiosamente los servidores con retrasos medios del orden de días en la obtención de imágenes son los que resultan ser los más rápidos en cuanto a descarga.

Desarrollando el algoritmo adecuado, que recorra las bases de datos de los distintos servidores, se puede priorizar las velocidades de bajada como criterio de decisión y en caso de resultados parecidos anteponer los tiempos de adquisición. Es por esto que la jerarquía para cada tipología de producto queda como representa la Figura 26.

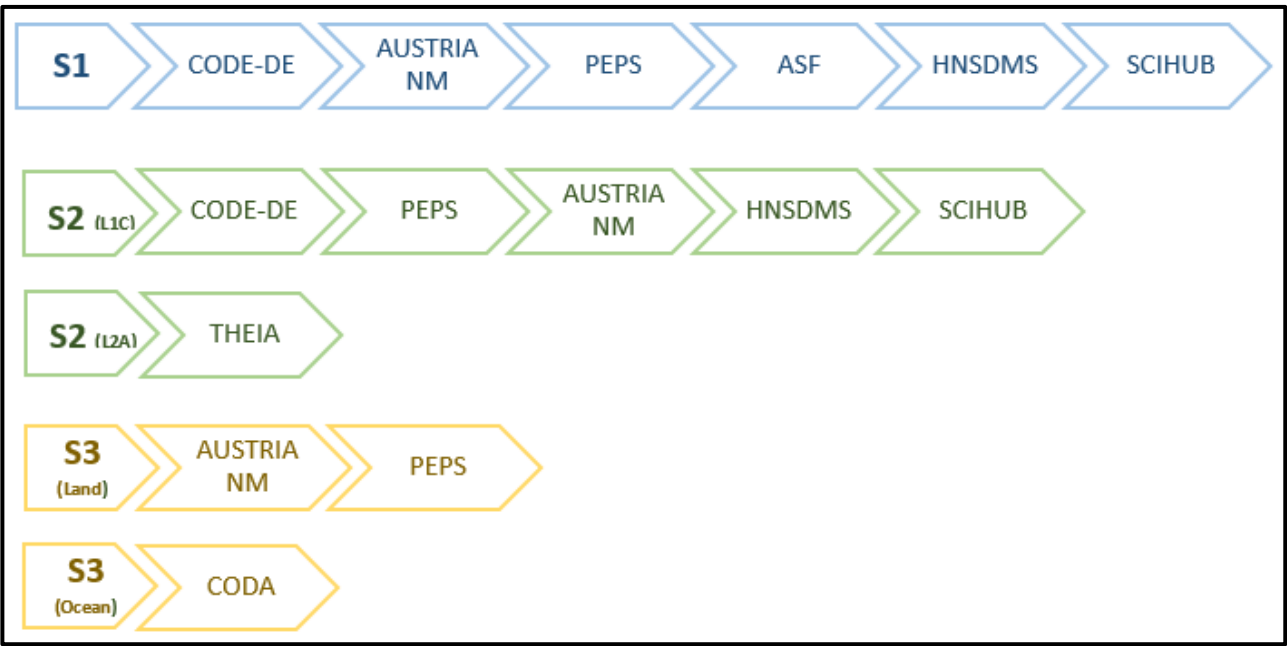


Figura 26: Diagrama de priorización de los repositorios para los distintos tipos de productos

De esta forma, aunque resulta poco probable por los resultados de adquisición para el S2 principalmente, si AUSTRIANM y CODE-DE tienen una imagen actual en un corto periodo de adquisición, se realizará la descarga pertinente siendo la más rápida de todos los repositorios posibles. El funcionamiento en concreto para optimizar la descarga se explicará en el apartado 4.3 y 4.4.

Con las conclusiones de los estudios se puede establecer a su vez un tiempo máximo de espera de descarga, de forma que, si el servidor en cuestión tuviera problemas no previstos y su velocidad descendiera notablemente, se procedería a la descarga desde el siguiente repositorio. Teniendo en

cuenta que uno de los objetivos es la ligereza del proceso, se estimarán de acuerdo a las medias estadísticas de tiempos obtenidos a las 6:00 las siguientes marcas máximas permitidas.

	ASF	AUSTRIANM	CODA	CODE-DE	HNSDMS	PEPS	SCIHUB	THEIA
Tiempos (s)	150	80	180	70	90	80	270	1000

Tabla 10: Límite estimado en cada servidor para el proceso de descarga de PyrenEOS

Las deducciones y soluciones obtenidas son las más eficaces para el entorno de producción del apartado 4.2.5 (lógicamente el escenario de evaluación es el de producción del proceso). Es por esto, que se ofrecen las herramientas de estudio de manera flexible para realizar dichas pruebas en las condiciones del entorno deseado, quedando a cargo las conclusiones del técnico responsable. A modo instructivo, se recomienda no cambiar los parámetros de los archivos BATCH, a excepción del perteneciente a la descarga desde AUSTRIANM por el límite de capacidad, ya que el objeto es el análisis del comportamiento de los servidores independientemente de las coordenadas o fechas de las imágenes.

4.3. DISEÑO Y DIAGRAMA DE FLUJO DEL PROGRAMA

Los resultados obtenidos en los estudios del apartado anterior, llevan al planteamiento del desarrollo de un sistema de descarga multiservidor, no sólo para las imágenes correspondientes a la zona geográfica de Navarra, si no para cualquier rango en coordenadas geográficas expresadas en grados polares. Con la palabra *multiservidor*, se quiere hacer referencia a la descarga de productos con un sistema jerárquico de servidores proveedores de imágenes, priorizando aquellos con mejores resultados, de tal forma que ante el fallo de cualquiera de estos por errores de conexión, bajas velocidades de descarga, mantenimiento del repositorio, etc, se obtengan los datos de los restantes servidores según el orden establecido.

La programación de la automatización de dicho proceso dependerá de los parámetros escogidos por el usuario, por ejemplo: sistema clasificatorio de almacenamiento, hora de descarga, tipología de productos, orden jerárquico de los repositorios, etc. A la hora de obtener una descarga eficiente, la lógica lleva a deducir el hecho que algunos de los parámetros citados no pueden establecerse por defecto, si no adaptarse al entorno donde se ejecutará esta herramienta. Para ello, el subsistema cuenta con los mecanismos explicados en la sección anterior, además de un procedimiento de registro de errores y logs que faciliten las tareas de mantenimiento del técnico encargado.

La plataforma de descarga se presenta en dos variantes, teniendo el usuario la posibilidad de elección, con iguales resultados finales. La distinción se da en la comodidad de uso frente al tiempo y grado de complejidad de instalación.

OPCIÓN A: Descarga automática con programación manual

Insertión de parámetros de manera manual en cada uno de los ficheros .bat existentes en las carpetas Check_DB, Downloaders y Web_Repositories_Test dentro de DOWNLOAD_SERVER. En estos archivos se indica de forma clara cuáles son los datos a cambiar. Ejemplo de ello queda ilustrado en la Figura 27, siendo recomendable dejar el resto de configuraciones tal cual se presentan. Para automatizar la descarga se debe crear manualmente una tarea programada por cada servicio que se desee, de acuerdo con el ANEXO A.

<pre>REM ===== REM User Parameters REM ===== set collection=S1 set maxtime=600 REM Define the Date-range to extract, or get it from command-line arguments: set starttime=2017-06-07 set finaltime=2017-06-09 REM Define the envelope to extract, or get it from command-line arguments: set lonmin=-2.2570 set lonmax=-2.2569 set latmin=42.779 set latmax=42.780</pre>	<pre>REM ===== REM User Parameters REM ===== set folder_tree=C/YYYY/MM/DD set zip_path=C:\SENTINEL\zip set unzip_path=C:\SENTINEL\unzip set collection=S1 REM ===== REM IP SERVER: change it REM ===== set ip_server=192.168.56.1</pre>
---	--

Figura 27: Ejemplos de las indicaciones y datos a cambiar por el usuario en los distintos .bat de DOWNLOAD_SERVER

En esta opción no se tiene un servidor web que proporcione un interfaz de visualización por lo que los errores y logs se deberán consultar en los ficheros que se crearán dentro del directorio Status_Files. Al no disponer esta solución de servidor web, la complejidad de la instalación disminuye, teniendo que seguir las indicaciones del ANEXO B y el apartado “Instalación de PostgreSQL” del ANEXO C, ya que sí se necesita esta base de datos, como se explicará posteriormente.

OPCIÓN B: Descarga automática con interfaz de usuario

En este caso se proporciona un interfaz web para la configuración de los distintos servicios, descarga o estudios, además de la visualización del estado del sistema. El manual de uso del interfaz se incluye en el presente trabajo en el ANEXO D.

La arquitectura de esta opción es cliente – servidor, resultando ser más cómoda y menos peligrosa a la hora de insertar los datos por parte del usuario ya que la asignación de estos se realiza en el lado del servidor de descarga de forma automática, a diferencia de la opción A donde el olvido de la actualización en algún fichero puede suponer fallos en alguno de los bloques de la descarga. Sin embargo, la instalación de este proceso es ligeramente más compleja que la opción anterior, teniendo que seguir las instrucciones dadas en el ANEXO B y ANEXO C por completo.

Comparando ambas posibilidades, se recomienda la instalación de la segunda, explicando desde este momento el desarrollo software y funcionamiento de este sistema.

A continuación se expone la arquitectura general de la plataforma correspondiente a la OPCIÓN B, ya que, en esencia, el procedimiento de descarga es el mismo y este último caso tiene mayor flujo de intercambio de datos.

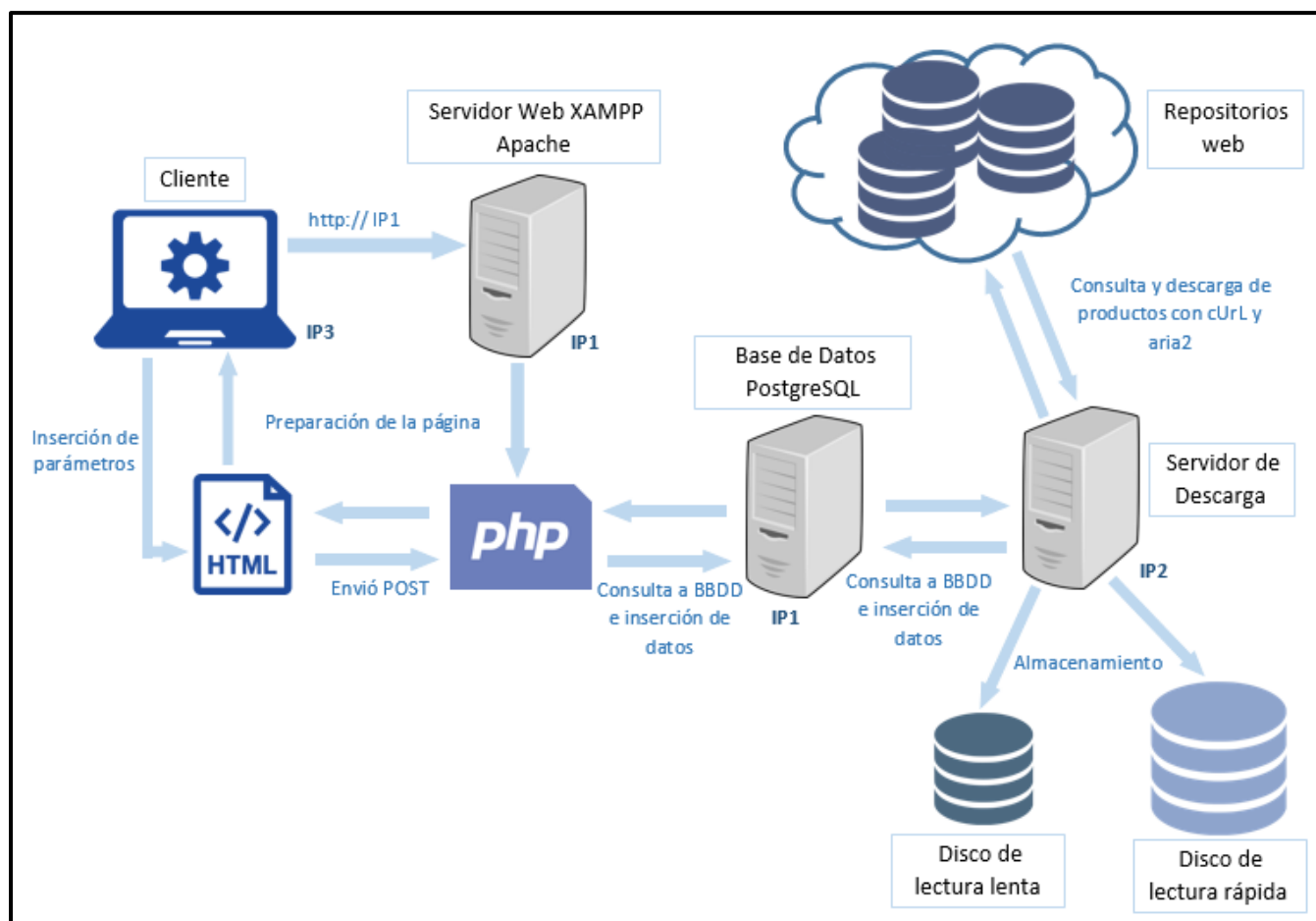


Figura 28: Arquitectura y flujo cliente-servidor del sistema de descarga

Las direcciones IP expresadas en la representación como IP1, IP2, IP3 consisten en direccionamiento IPv4 dentro de una misma subred, recordando que se está simulando, en la medida de lo posible, la red privada de Gobierno de Navarra, siendo la red 192.168.56.0/24 para el entorno en el que se ha evaluado la aplicación. Como es apreciable, el servidor web y la BBDD PostgreSQL [38] tienen la misma dirección de red, esto es debido a que en el entorno simulado estos dos elementos se encuentran en la misma máquina virtual, reiterando la Figura 17 en el apartado 4.3.1.

Siguiendo en la línea del servidor web y la base de datos, se ha escogido como primero el servidor XAMPP 5.6.30 [39] al ser independiente, de código libre y gratuito, tener una manejabilidad sencilla y sobre todo ser un servidor multiplataforma, es decir, funcional en otros sistemas operativos como por ejemplo Linux, lo cual es un punto a favor de cara a líneas futuras del programa. Contiene el propio servidor web Apache, PHP 5.6.30 y base de datos MySQL, admitiendo extensiones.

Por otro lado, se tiene como base de datos PostgreSQL 9.5.6, un sistema de gestión de datos objeto-relacional multiplataforma, con código fuente disponible libremente y que muestra muchas semejanzas a SQLServer. Se trata de una base de datos que presenta un comportamiento satisfactorio ante grandes cantidades de datos de distintos tipos, otorgando además de facilidad para su

administración. Centrando las ventajas de este en el ámbito del proyecto PyrenEOS, cuenta con el plugin PostGIS [20] que añade analítica geográfica a las tablas, manejando geometrías y realizando cruces espaciales. Esto es necesario para la carga de datos de imágenes satélites y la obtención de datos por parte de herramientas SIG (Sistema de Información Geográfica) para la visualización y edición de datos geoespaciales a través de peticiones WMS (Web Map Service), como GeoServer [21], el segundo subsistema de la plataforma PyrenEOS.

Finalizando con el análisis de esta arquitectura, se puede comprobar cómo no existe un contacto directo entre el cliente y el servidor de descarga, siendo a su vez imposible el acceso a las imágenes almacenadas en los diferentes discos. El intermediario de esta comunicación es el servidor que contiene la base de datos, recibiendo consultas y datos y enviando respuestas por parte de ambos sistemas. Si bien puede parecer una buena solución, no será hasta la puesta en producción cuando se compruebe realmente la eficiencia de este diseño. Para ver mejor el flujo del programa a nivel general en el programa diseñado, se presenta seguidamente la Figura 29.

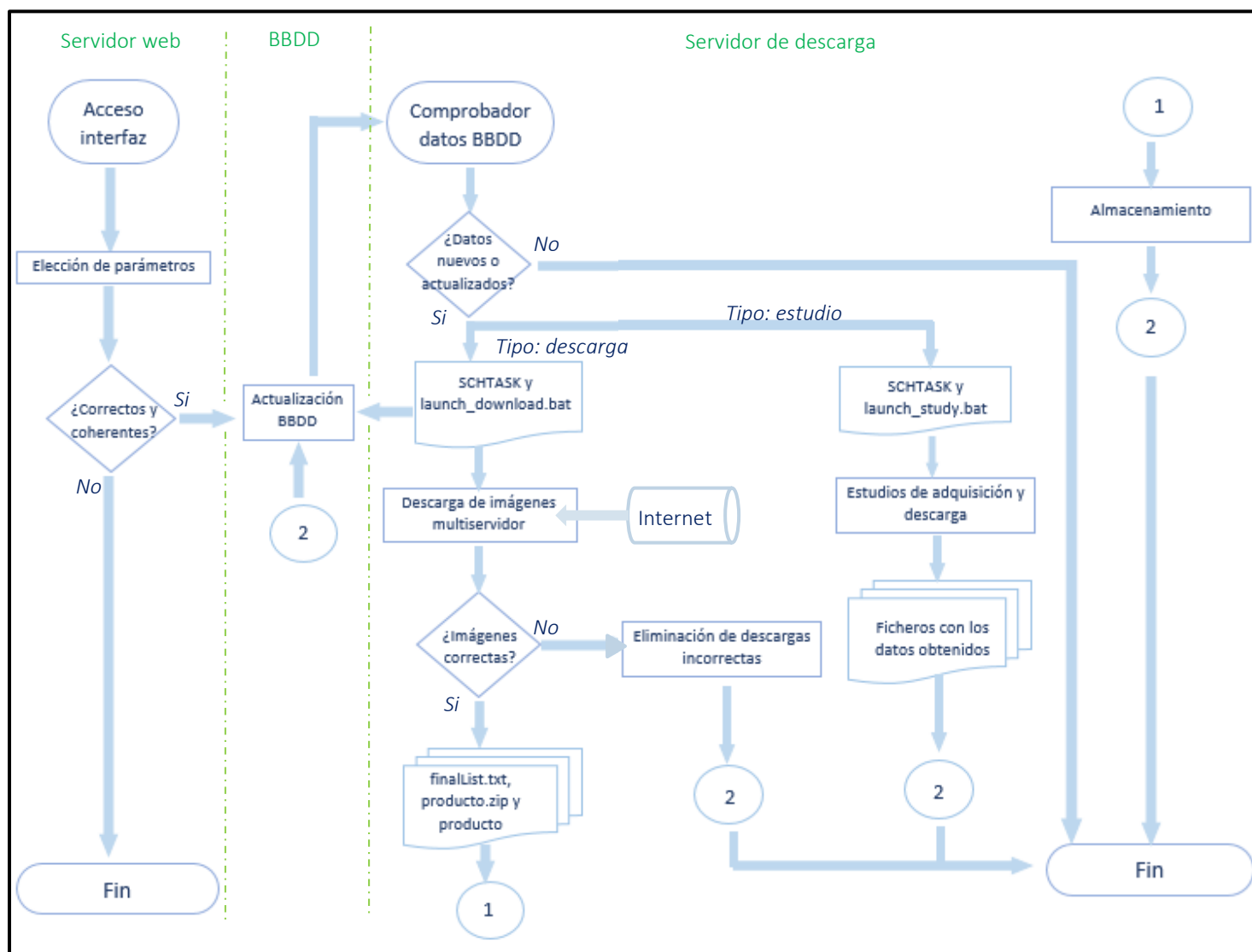


Figura 29: Diagrama de flujo general entre servidor web, base de datos y servidor de descarga

El diagrama presentado tiene como objetivo dotar de una visión en conjunto para conocer como se da el intercambio de flujos entre los distintos elementos de la aplicación diseñada, en el apartado 4.4 se explicará más detalladamente cada uno de estos bloques.

Como es apreciable, el servidor de descarga contiene la mayor parte de los procesos para realizar los distintos servicios: descarga de productos como tal o estudios de variables de entorno. Sin embargo, no se llevan a cabo hasta que los datos necesarios se encuentren en la base de datos. Siguiendo el diagrama en sentido contrario, esto quiere decir que el proceso entero se iniciará cuando el usuario escoja los parámetros requeridos para ello, teniendo que pasar por un filtro que determine si son correctos y coherentes, insertándolos en la base de datos.

Sean cuales sean los resultados, el servidor de descarga cuenta con un módulo de registros, es decir, tanto para fallos como para éxitos del sistema se realizan registro de estos, insertándolos y actualizando la base de datos. Es notable, por tanto, la importancia de la existencia de esta base de datos no sólo como mecanismo de almacenamiento, sino como método de monitorización. De esta forma, se logra que el servidor web se percate del estado del proceso de descarga y el servidor propio de descarga conozca el estado del anterior. Este sistema de gestión de datos se trata, además del nexo entre apache y el descargador, el elemento de comunicación entre el subsistema de descarga y el subsistema de geoprocso del proyecto PyrenEOS.

4.4. DESARROLLO DEL PROGRAMA

4.4.1. Interfaz web de Configuración y Monitorización

La página creada con nombre Sentinel Product Ingestor consiste en un interfaz de usuario que facilite la configuración y monitorización del sistema de descarga de imágenes de satélites Sentinel. El manual de uso de esta aplicación web viene adjunta en el presente trabajo en el Anexo D, además de encontrarse en la página principal de la web. Se trata de un interfaz gráfico que realiza consultas a la base de datos y representa los resultados de estas en la web.

Debido a su poca criticidad, los lenguajes utilizados para su creación son HTML5, CSS3, PHP5 y Javascript, no necesitando ningún tipo de Framework, ya que está destinado a su uso en puestos de trabajo fijos, es decir en navegadores de ordenadores y no de terminales móviles. Sin embargo, que sea sencilla de implementar no quiere decir que no dote de cierta inteligencia, contando con algoritmos y funciones de comprobación y filtrado de variables en el lado cliente.

No se trata de una web abierta a todo tipo de usuario, en este caso al trabajador de Gobierno de Navarra, siendo necesario estar en posesión de credenciales para su acceso. Es por esto que se utilizan sesiones en el lado del servidor mediante PHP. El punto más importante de este apartado, es la distinción de tres tipos de usuarios según el rol asignado con diferentes permisos. Los privilegios de cada uno vienen descritos en la Tabla 11.

Rol	Añadir usuarios	Configurar rutas de almacenamiento	Configurar parámetros de descarga	Lanzar estudios de adquisición y velocidad	Visualización de registros de estado
admin	X	X	X	X	X
expert			X	X	X
normal					X

Tabla 11: Privilegios según el tipo de usuario en Sentinel Product Ingestor

Extrayendo información de la tabla superior, se deduce que el usuario tipo admin será el único que pueda crear usuarios pudiendo escoger únicamente entre usuario con rol experto y normal. Por tanto, el administrador del sistema de descarga como tal es único.

Cabe mencionar que la distinción de usuarios no implica independencia en las cuentas de cada uno, lo que quiere significar que las configuraciones que realice un usuario las podrá visualizar otro y viceversa.

Terminando con esta primera parte explicativa, pese a poder ser trivial su significado, la configuración de rutas contendrá el formulario a rellenar donde se deberá indicar el sistema de clasificación y las unidades de almacenamiento para lectura rápida y lenta. Se consideran parámetros importantes por lo que su acceso se restringe únicamente al técnico administrador. En la configuración de variables de la descarga de imágenes, se deberán indicar, por cada tipo de producto, las coordenadas geográficas, rangos de fechas y hora programada para su descarga y, opcionalmente, el orden jerárquico de los repositorios web y sus tiempos máximos de descarga. De manera similar ocurre en los estudios de adquisición y descarga pero con horas predefinidas. No se trata de una elección inmutable, ya que es posible la eliminación o la parada de los procesos.

Pasando a la parte más técnica y continuando con las secciones web del párrafo anterior, el lado cliente cobrará importancia al tener implementados mediante Javascript funciones para la comprobación de parámetros correctos y coherentes, por ejemplo, ninguno de ellos nulo, números decimales con punto, coordenadas dentro de los rangos establecidos, orden de los servidores sin repeticiones, etc. De esta forma, se notifica al usuario de las correcciones a realizar antes de enviar ningún tipo de dato al lado servidor mediante el método POST, que oculta los nombres y valores de las variables no representándolas en la URL.

Una vez recibidos los datos por el PHP, éste deberá ejecutar unas funciones u otras dependiendo del estado de la base de datos y de la petición llegada desde el cliente, siendo para cada tipo de configuración un árbol de decisión de menor o mayor extensión.



A modo general, cada PHP de estos apartados se guiará por el siguiente diagrama:

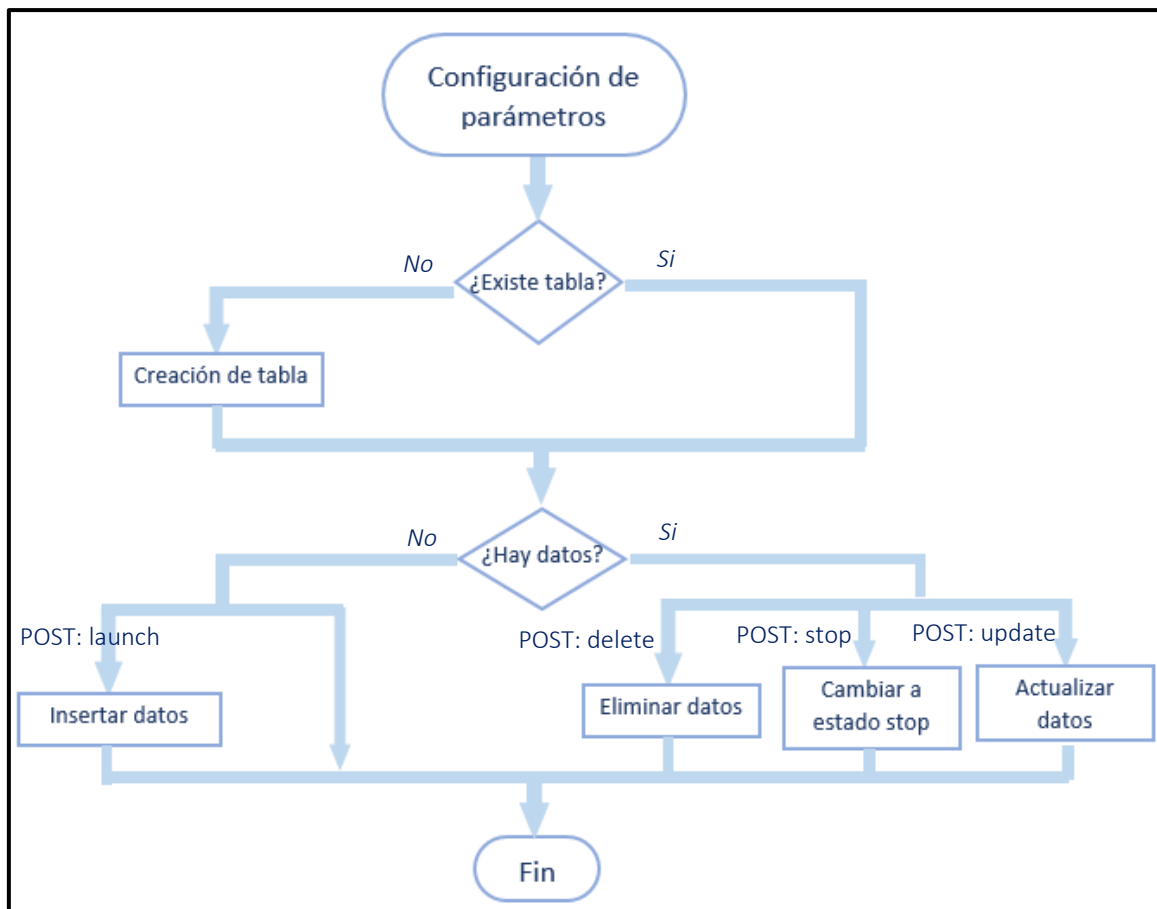


Figura 30: Diagrama de decisión de los php dedicados a las configuraciones del sistema de descarga

A la hora de conectarse a la base de datos y realizar instancias relacionada con esta, el PHP utiliza funciones propias para PostgreSQL, como por ejemplo *pg_connect* para establecer la conexión, *pg_query* para realizar la consulta, *pg_fetchall* para obtener el array de la consulta, *pg_escape_string* para evitar caracteres peligrosos, etc. [40]

Es importante en estos archivos la creación automática de las tablas necesarias, siempre y cuando esta no exista. Ejemplo de la declaración SQL necesaria para ello se ilustra en la Figura 31. De la función, se puede concluir también que no sólo se necesita el enunciado SQL para formar la tabla, si no también otra correspondiente a la creación y asignación de una clave primaria que se complete de forma automática a través de una secuencia.

El interfaz de usuario tendrá, por tanto, una labor muy importante para poder iniciar el sistema de descarga del proyecto PyrenEOS.

```

function createTable($dbcnxconfig){
    $sequence="CREATE SEQUENCE config_tools_id_seq";
    $createseq=pg_query($dbcnxconfig,$sequence);
    $query="CREATE TABLE public.config_tools
    (
        id integer NOT NULL DEFAULT nextval('config_tools_id_seq'),
        name character varying NOT NULL,
        type character varying NOT NULL,
        status character varying NOT NULL,
        start_date date NOT NULL,
        end_date date NOT NULL,
        lonmin character varying NOT NULL,
        lonmax character varying NOT NULL,
        latmin character varying NOT NULL,
        latmax character varying NOT NULL,
        CONSTRAINT scl PRIMARY KEY (id)
    )
    WITH (
        OIDS=FALSE
    );
    ALTER TABLE public.config_tools
        OWNER TO user_sentinel;
    ALTER SEQUENCE config_tools_id_seq OWNED BY public.config_tools.id;";
    $create=pg_query($dbcnxconfig,$query);
}
  
```

Figura 31: Declaración SQL para la creación de una tabla a través de php

4.4.2. Descarga de Parámetros y Automatización

Este apartado consiste en el primer bloque del programa, el cual dará inicio a la descarga de imágenes y/o estudios de variables de entorno. En concreto, es el script encargado de la automatización del sistema, denominado `check_db.py`, creando y configurando, mediante líneas de comandos de Windows, las tareas oportunas y los archivos que se ejecutarán como acciones en estas.

El desarrollo de este bloque se ha realizado mediante el lenguaje de programación Python, estructurando el código en funciones a fin de conseguir mayor claridad y comprensión. No obstante, el punto más significativo es la importación del módulo `psycopg2` [41], una librería en Python cuya función es la conexión del propio script con la base de datos PostgreSQL. A partir de este apartado, las restantes secciones del capítulo 4.4 se desarrollarán con las mismas herramientas. El uso correcto de esta librería se rige en los siguientes pasos:

- Establecimiento de la conexión: llamando a la función `connect` del módulo `psycopg2` con argumentos la dirección IP de la base de datos, el nombre de la propia base y las credenciales de usuario.
- Creación del cursor: mediante el método que posee la conexión denominado `cursor`, se abre esta estructura de control cuyo fin es el de recorrer, insertar o actualizar filas en una tabla.
- Envío de query: con el anterior establecido, se puede ejecutar consultas a la base de datos mediante declaraciones SQL, siendo estas los argumentos de la función `execute` del cursor.
- Commit de los cambios: en consultas que impliquen inserción o cambio de datos en una tabla por petición de cliente, en este caso el script de Python, se deben remitir dichas actualizaciones a la base de datos. Para ello se invoca a la función `commit` que posee el cursor.
- Cierre de conexión: al terminar con las consultas sobre la misma base de datos la conexión tiene que cerrarse para liberar recursos, la forma de realizar este paso será a través de la función `close` de la conexión.

```
try:
    #Connection to database#
    connection = psycopg2.connect(host="%s"%(options.ip_server), database="configuration_db", user="user_sentinel", password="sentinel_pass")
    cursor = connection.cursor()
    #-----#
    sequence="CREATE SEQUENCE logs_records_id_seq"
    cursor.execute(sequence)
    connection.commit()
except psycopg2.DatabaseError, e:
    print 'Error %s'%e
    sys.exit(-1)

finally:
    if connection:
        connection.close()
```

Figura 32: Ejemplo del uso correcto de la librería `psycopg2` para la conexión a la base de datos

Retomando el primer párrafo y con los conocimientos del segundo, a continuación se expone el diagrama de flujo de este script para lograr la automatización del sistema.

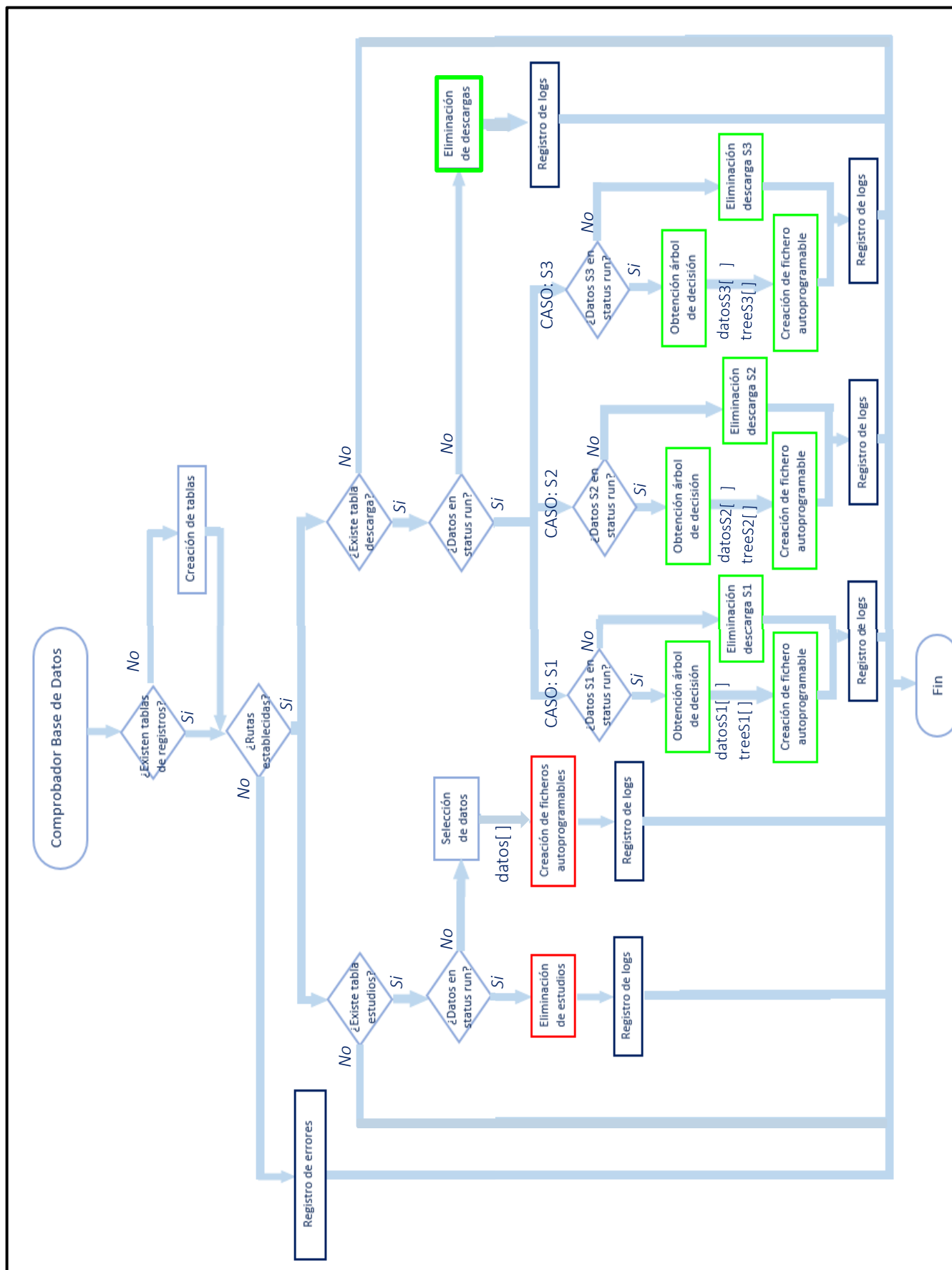


Figura 33: Diagrama de flujo del bloque de Descarga de Parámetros y Automatización, script check_db.py

De la anterior representación, se aprecia que este script se basa en múltiples condiciones para la obtención de un resultado u otro y, como es lógico, cuantas más líneas de código a procesar más se ralentiza el proceso. Además, la ejecución de este bloque se realiza cada 10 minutos las 24 horas del día, a través de su programación mediante el asistente de Windows como se explica en el Anexo B. Es por todo esto que se establece un segundo archivo, `launch_check.vbs`, cuyo fin es ocultar el terminal de Windows cada vez que se pone a correr este proceso, ejecutándolo en segundo plano.

Las creaciones de la automatización de los estudios y las descargas son muy parejas en una primera visión general, como se puede ver en la Figura 33. Sin embargo, tienen pequeñas distinciones. Es por esto que en la ilustración se hacen dos claras diferenciaciones con los colores verde y rojo, que se traducen en la creación de dos módulos Python a fin de concebir un código entendible: `check_tools.py` para las funciones correspondientes a los estudios, en color rojo, y `check_download.py` para las descargas, en color verde.

En la siguiente ilustración, se esquematiza el flujo de las dos funciones del módulo de estudios invocadas en la Figura 33.

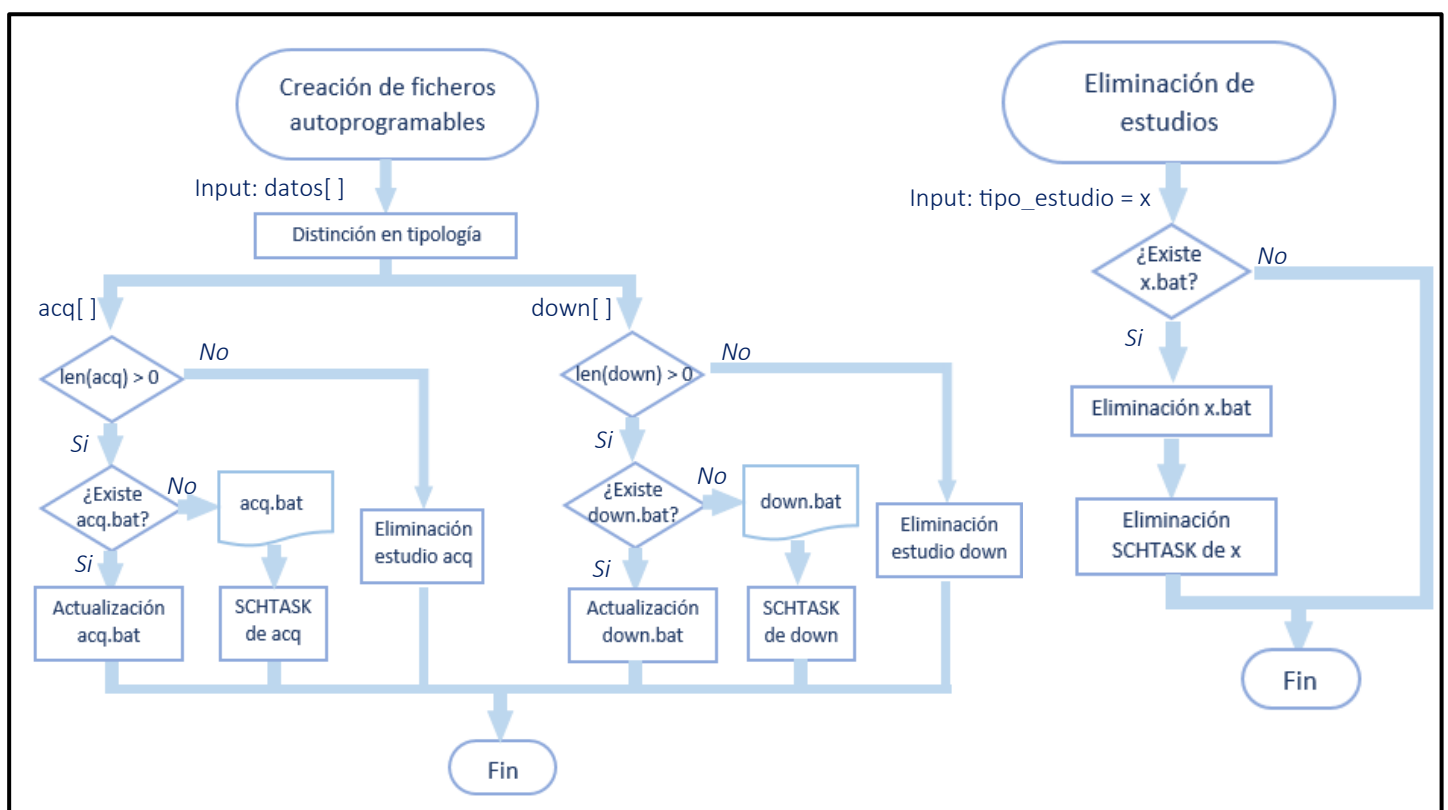


Figura 34: Funciones para la automatización de estudios en `check_tools.py`

Centrando la explicación en el método de producción, sea cual sea el tipo de estudio a realizar, se obtiene un fichero en Batch con las líneas de comandos necesarios para el lanzamiento de los distintos repositorios con los parámetros escogidos por el usuario a través del interfaz. La estructura en la que está planteada el código de este fichero, hace posible la ejecución de los estudios de manera paralela. Estos estudios y sus resultados se dan por explicados en la sección 4.2.4.

La hora a la que se lanzará la tarea que contiene como acción dicho archivo será la siguiente hora en punto más próxima a la actual, significando *actual* la hora en la que se crea el fichero. Esta tarea se pondrá a correr cada 60 minutos para el caso de estudios de descarga y cada 15 minutos para el de adquisición. A diferencia de apartados anteriores, la programación de la tarea se realizará de forma automática con el comando *SCHTASKS /Create [opciones]*. [42]

Por último, se muestra el diagrama de flujo de las funciones del módulo de descargas invocadas en el esquema general, donde se pueden apreciar las similitudes con la Figura 34.

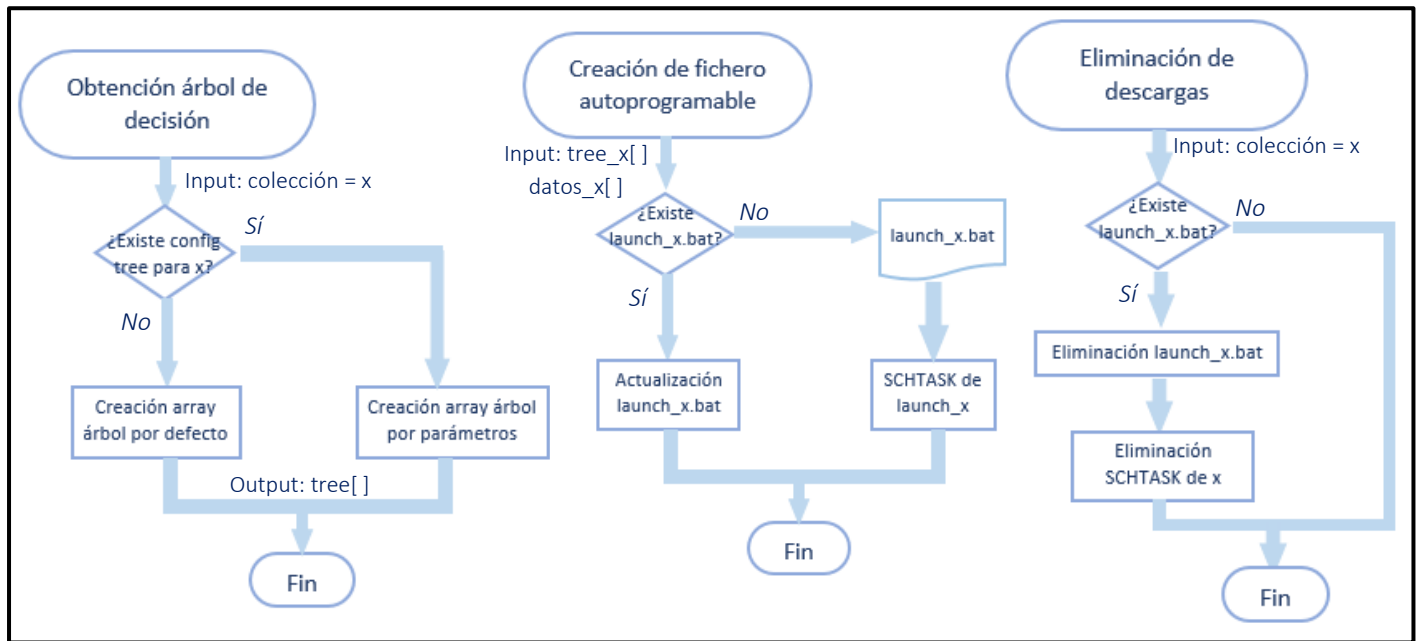


Figura 35: Funciones para la automatización de descarga en *check_download.py*

La primera diferencia visible entre esta y la representación anterior es el hecho de que las distinciones entre tipología de datos, según tipo de estudio o colección, se realizan en el primer caso dentro de la función *Creación de ficheros autoprogramables* y en el segundo antes del uso de esta.

La producción de la descarga pasa en este caso por dos funciones: *Obtención del árbol de decisión*, correspondiente al orden y tiempos máximos de los repositorios de imágenes, y la *Creación del fichero autoprogramable*. El archivo resultante de este último método difiere en cuanto a estructura respecto al de los estudios, y es que la ejecución de la descarga de una misma colección se realiza de forma secuencial, pero de manera paralela entre distintas colecciones. De este modo, hasta que no finalice la descarga desde el primer servidor, no se continuará con el siguiente y así hasta terminar con el proceso de descarga como tal e iniciar a continuación el de almacenamiento.

La hora a la que se ejecuta por primera vez la tarea programada, de nuevo mediante SCHTASKS, es la escogida por el usuario con fecha el mismo día, si la hora actual es menor que la dada, o el día siguiente, si la actual es mayor.

Como conclusión de este apartado, cabe destacar la importancia de la función de creación en ambos casos, ya que es en este proceso en el que la plataforma se hace multiservidora y automática.

4.4.3. Descargador de Imágenes Multiservidor

A fin de lograr una plataforma de descarga multiservidor, no sólo basta que el fichero autoprogramable secuencie el proceso entre los repositorios, si no que éstos se deben estructurar de la misma manera para que no se produzcan incompatibilidades entre resultados. Con esto, se presenta a continuación el diagrama de flujo por el que se rigen los distintos servidores.

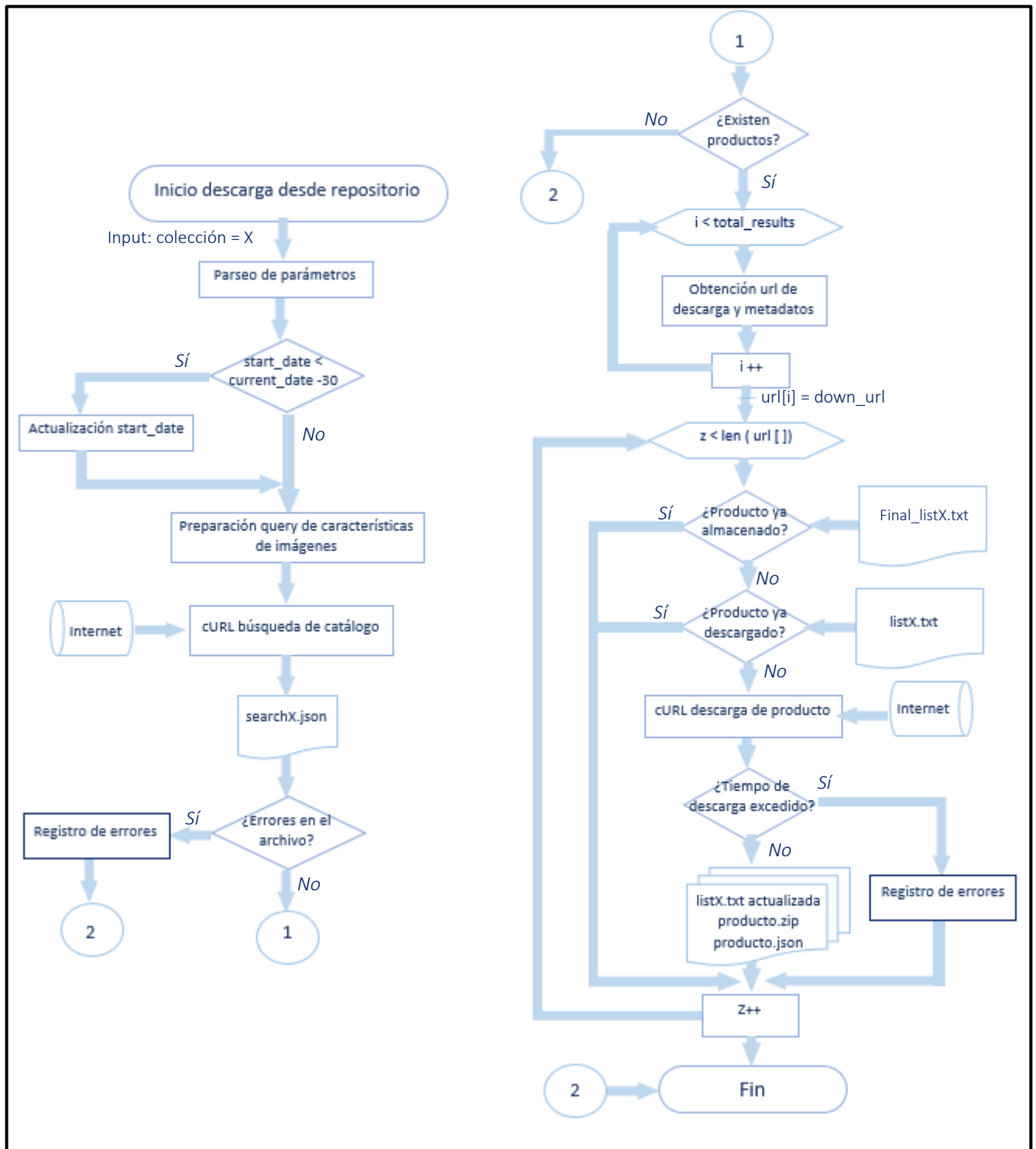


Figura 36: Diagrama de flujo del proceso de descarga desde un repositorio

Con propósito de aclarar la información representada en la Figura 36, a continuación se explica de manera resumida el proceso de descarga.

En primera instancia, debido al límite de resultados totales a devolver que cada repositorio establece, se debe comprobar que la fecha de inicio sea relativamente actual. En caso negativo, se cambia dicha fecha por otra más reciente a fin de no tener un error por rango de resultados excedido.

Con las características de las imágenes requeridas establecidas (coordenadas, fechas...), se procede mediante cURL a la obtención del catálogo de imágenes que cumplen dichos requisitos y se encuentran en la base de datos del repositorio. Es importante la comprobación del contenido de este, de manera que si es correcto, es decir, existen datos en formato JSON, se podrá continuar con el proceso.

A partir de aquí, con el archivo anterior, se añade a un array de productos las características de dichas imágenes, de manera que para cada una el índice sea el nombre del producto y el valor otra cadena que contiene la url de descarga, el tamaño y los metadatos más importantes.

Por cada producto del array, se debe comprobar si esa imagen se encuentra ya almacenada. Esto se conoce mediante la lectura del fichero *Final_listcolección.txt*, el cual contiene los nombres de los archivos ya guardados en la unidad de almacenamiento del repositorio que se está creando. De no encontrarse, se tiene que realizar lo mismo pero esta vez con el fichero *listcolección.txt*. A diferencia del anterior, éste contiene los nombres de los productos descargados, en el momento de ejecución de la tarea, de los servidores antecesores al actual. De esta manera, no se duplica ninguna descarga y se consiguen las imágenes adquiridas en el menor tiempo posible, ya que se comprueban las bases de datos de todos los repositorios web.

Si la descarga se cumple con éxito, es decir, el tiempo de esta no ha sobrepasado el límite establecido por el usuario (opción *max-time* en el comando *curl*), se guardará el producto en una carpeta provisional en la propia máquina servidora de descarga. Además se actualizará el fichero de descargas del proceso actual y se creará un JSON con los metadatos de la imagen, el fin de este último se verá en el apartado 4.4.5.

Este bloque del programa junto al explicado en la sección anterior, 4.4.2, conforman los pilares de la plataforma de descarga para el proyecto PyrenEOS.

4.4.4. Comprobador de Imágenes Corruptas

El desarrollo de este apartado se produce con el fin de evitar el almacenamiento de imágenes corruptas, entendiendo como *corruptas* aquellas con falta de ficheros (o añadidos) debido a una descarga incompleta o pérdida de datos en la propia imagen. Este elemento se incluye en el proceso de descarga, dentro del script de Python *unzip_and_store.py* y el módulo *control*, importado en este, con la función destinada a la comprobación de errores en las imágenes.



Como se veía en la sección 4.2.3 cuando se realizaron los primeros descartes, se comentó que el repositorio de Amazon tenía una ventaja con respecto al resto, poseía un algoritmo de filtrado por el que no adquiría aquellas capturas que no pasaban los niveles de calidad según el proceso OLQC.

Este problema se sigue manteniendo, por lo que se hace necesario, a fin de conseguir una plataforma eficiente, implementar un algoritmo por el que se compruebe si existen errores o no en dichas imágenes.

OLQC (Online Quality Control) solo se da en los productos procedentes del Sentinel-2. Acudiendo al manual S2_MSI_Product_Specification, se da a conocer cómo encontrar este tipo de error. El proceso de control, en caso de fallo, comprueba 5 tipos de errores: Calidad General, Calidad Geométrica, Calidad Radiométrica, Calidad del Sensor y Corrección del Formato. Si se desea conocer más acerca de los datos comprobados por cada tipo véase el manual en [43].

De acuerdo con este documento, este proceso cuando obtiene un fallo genera un informe dentro de la carpeta GRANULE > QI_DATA y DATASTRIP > QI_DATA. Sin embargo, existe otra variante más inmediata. En el directorio principal, se encuentra un archivo con nombre MTD_MSIL1C.xml con características generales de la propia imagen. Entre todas, interesa el elemento Quality_Control_Checks, cuya información se representa mediante un ejemplo en la Figura 37.

```
-<Quality_Control_Checks>
  -<Quality_Inspections>
    <SENSOR_QUALITY_FLAG>PASSED</SENSOR_QUALITY_FLAG>
    <GEOMETRIC_QUALITY_FLAG>FAILED</GEOMETRIC_QUALITY_FLAG>
    <GENERAL_QUALITY_FLAG>PASSED</GENERAL_QUALITY_FLAG>
    <FORMAT_CORRECTNESS_FLAG>PASSED</FORMAT_CORRECTNESS_FLAG>
    <RADIOMETRIC_QUALITY_FLAG>PASSED</RADIOMETRIC_QUALITY_FLAG>
  </Quality_Inspections>
  -<Failed_Inspections>
    -<Datastrip_Report datastripId="S2A_OPER_MSI_L1C_DS_MPS_20170317T003712_S20170316T230521_N02.04">
      -<REPORT_FILENAME>
        DATASTRIP/DS_MPS_20170317T003712_S20170316T230521/QI_DATA/GEOMETRIC_QUALITY.xml
      </REPORT_FILENAME>
    </Datastrip_Report>
  </Failed_Inspections>
</Quality_Control_Checks>
```

Figura 37: Estado del control de calidad en el archivo MTD_MSIL1C.xml

El algoritmo propio para la plataforma tiene la función de buscar este elemento y recorrer los valores de Quality_Inspections. Este y la comprobación de que el tamaño del zip se corresponde con el tamaño indicado en el metadato del json extraído en el apartado anterior, se representa en la Figura 38. Para lograr esto de manera más fácil, se importa la librería xml.etree.ElementTree, útil para la lectura de ficheros con extensión xml.

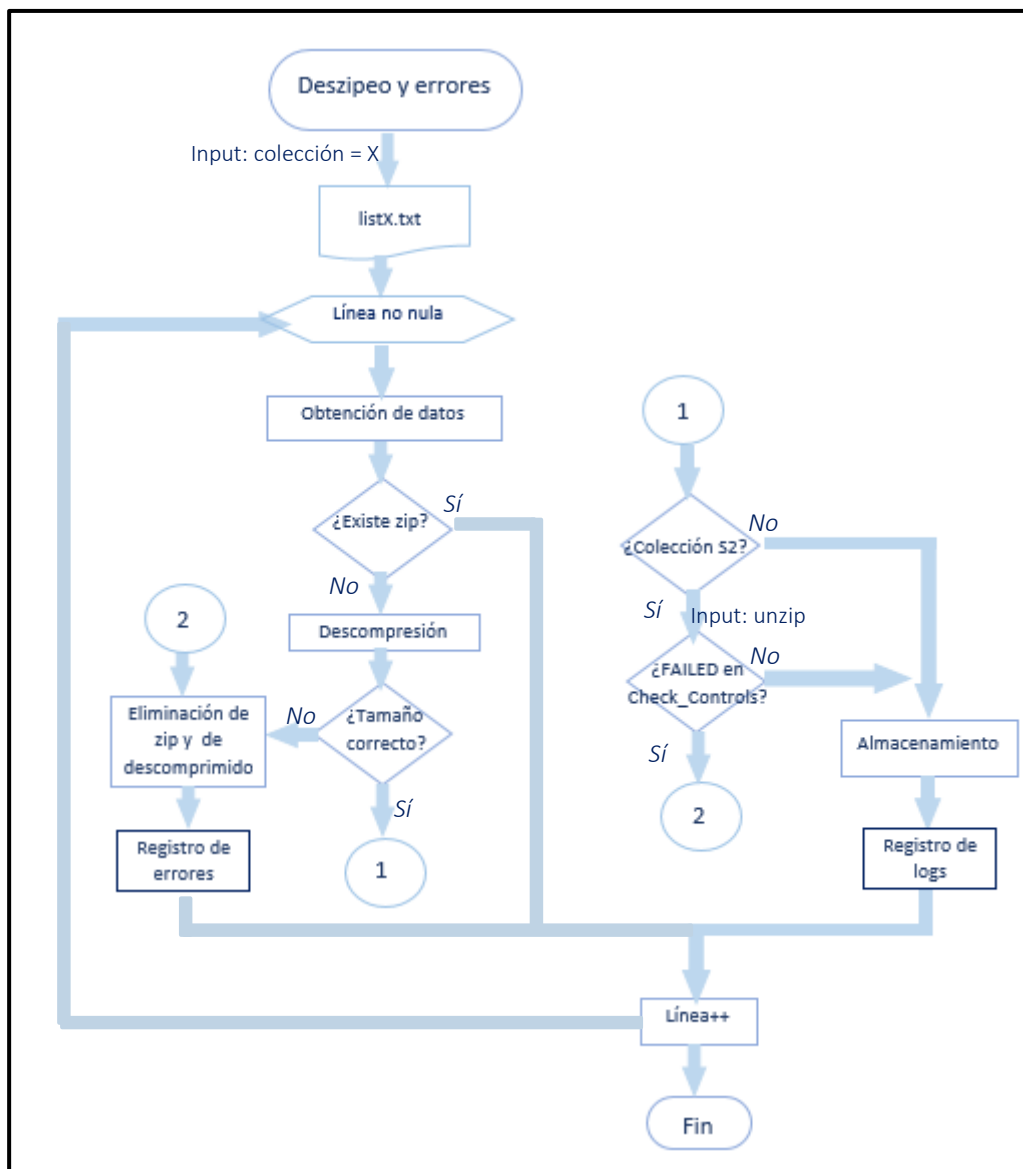


Figura 38: Diagrama de flujo de la descompresión y comprobación de errores de los productos

4.4.5. Carga a Base de Datos PostGIS y Almacenamiento

Concluyendo con el proceso de descarga de la plataforma, se debe proceder a su almacenamiento y correspondiente carga a la base de datos con la extensión de PostGIS, sentinel_data_db.

El proceso de actualización en la base de datos se realizará de manera convencional, con la herramienta psycopg2 y su correcto uso, como quedaba explicado en el apartado 4.4.2. La lógica que seguirá será la creación de la tabla *colección*json en caso de no existir, y la inserción del nombre y metadatos de la imagen correspondiente.

Cabe mencionar la existencia de una herramienta más potente y eficaz para realizar cargas de datos a una base PostGIS: ogr2ogr [44]. La ventaja principal que posee es el hecho de que es capaz de comprobar la existencia o no de una tabla y su correcta creación de forma automática con tan sólo introducir el fichero JSON de los metadatos. Esta solución habría sido la más idónea, sin embargo durante el periodo de producción del presente Trabajo Fin de Grado, no se ha logrado con éxito su implementación.

Los ficheros existentes en Data_provisional, son movidos a las unidades y/o rutas escogidas por el usuario para su almacenamiento a través del interfaz web. El sistema de clasificación de carpetas estará determinada también por este. Para lograr este proceso de cambio de destino, se importa la librería *shutil*, la cual permite operaciones a alto nivel sobre archivos y directorios. Los distintos tipos de organización disponibles a escoger son:

- Colección/Año/Mes/Día
- Año/Mes/Día/Colección
- Año/Mes/Día

El módulo desarrollado *store* contiene las funciones que permiten la creación de dichos directorios y extracción de fechas para cada producto, ya que éstas últimas son parte de la nomenclatura de cada imagen.

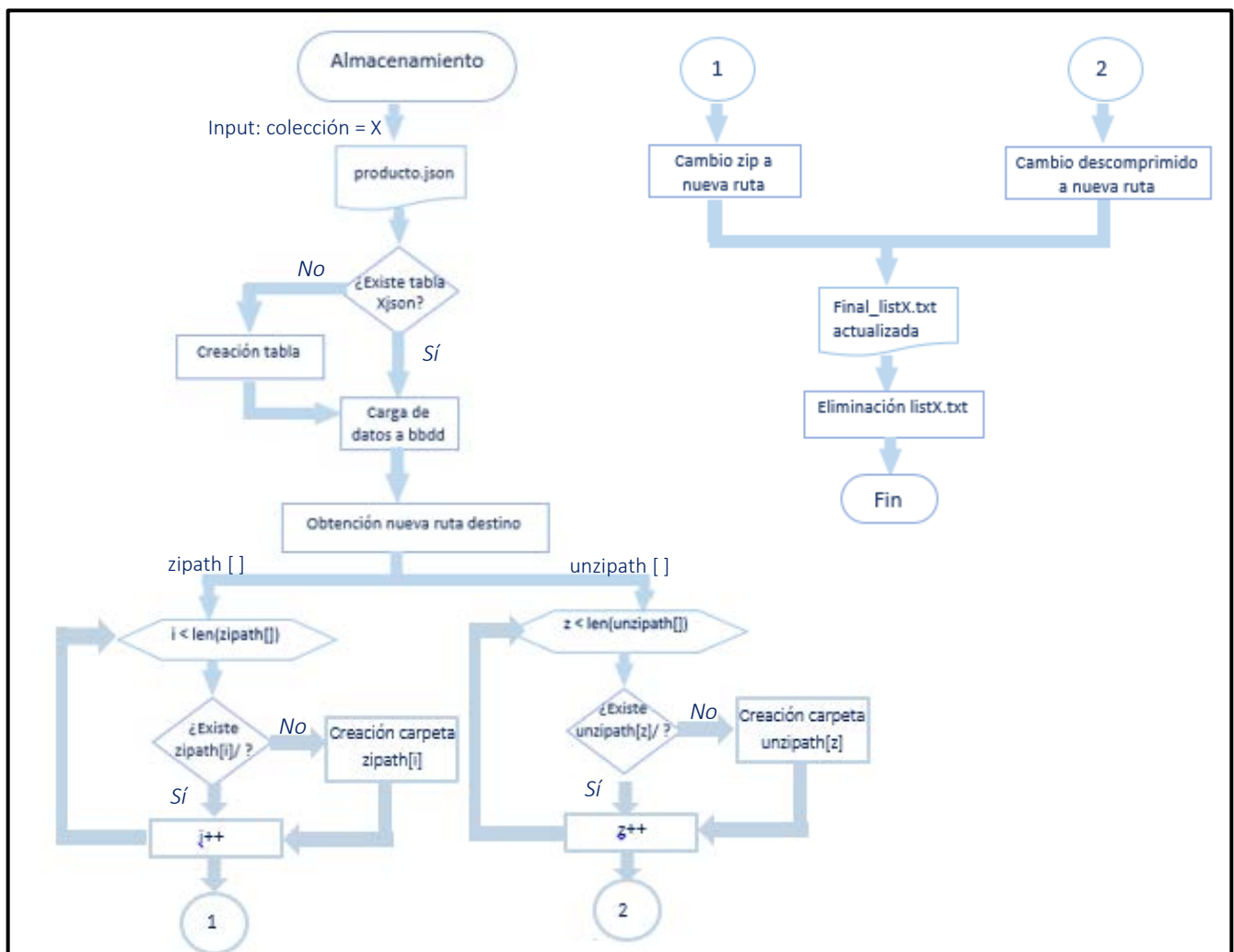


Figura 39: Diagrama de flujo de almacenamiento en disco y base de datos de los productos

4.4.6. Seguimiento de Estado y Errores

En los anteriores apartados correspondientes a la sección Desarrollo del Programa, se ha podido ver como en sus diagramas de flujo existían procesos con el nombre de *Registro de errores* y *Registros de logs*. El objetivo de este último punto es el de explicar estos dos métodos.

La mayoría de los scripts explicados cuentan con la importación de un módulo denominado `register_record.py`, en el que se encuentra la función `write`. Este método realiza dos procesos: escribir el tipo de registro y el comentario en el fichero de estados correspondiente, dentro de `Status_Files`, y hacer lo propio pero insertándolo en la tabla de records de la base de datos mediante `psycopg2`.

Considerando el esquema de esta función comprendida con la explicación anterior, se expone, para concluir, una tabla con los tipos de registros captados por el programa.

TIPO	ASUNTO	PROCESO
Error	Rutas de almacenamiento no establecidas	Descarga de parámetros y automatización
Log	Tarea_estudio creada, preparada para lanzar a las X horas	Descarga de parámetros y automatización
Log	Tarea_estudio eliminada, proceso finalizado	Descarga de parámetros y automatización
Log	Tarea_descarga creada, preparada para lanzar a las X horas	Descarga de parámetros y automatización
Log	Tarea_descarga eliminada, proceso finalizado	Descarga de parámetros y automatización
Error	Respuesta nula desde el servidor X	Descarga de catálogo desde un servidor
Error	ErrorCode: 400 – 500 en el servidor X	Descarga de catálogo desde un servidor
Error	SyntaxError en la respuesta del servidor X	Descarga de catálogo desde un servidor
Error	ValueError en la respuesta del servidor X	Descarga de catálogo desde un servidor
Error	El servidor excedió el tiempo máximo de descarga permitido	Descarga de producto desde un servidor
Error	Producto descargado incompleto	Comprobación de errores
Log	Producto correctamente descargado desde servidor X	Comprobación de errores
Log	Producto almacenado correctamente	Almacenamiento

Tabla 12: Tipos y asuntos de registros notificados por el programa de descarga

De esta manera, se constituye una herramienta que, gracias a estas notificaciones, aporta un punto de comodidad a la hora de monitorizar el sistema.



4.5. FUNCIONAMIENTO Y RESULTADOS

4.5.1. Puesta en marcha de la aplicación

La puesta en producción en el entorno tecnológico de Gobierno de Navarra del software diseñado no ha sido posible en el periodo de desarrollo de este. Esto ha sido debido al largo periodo de tiempo que se da entre las diferentes reuniones, en las cuales al ser PyrenEOS un proyecto reciente, se presentaban las bases a hacer pero no se llegaba a acuerdos fijos. Esto se traducía a la existencia de dudas por parte del Servicio de Infraestructuras y Centros de Soporte, no consiguiendo disponer de un servidor de pruebas físico para la comprobación de la aplicación.

Al no poder poner en marcha la plataforma sobre una arquitectura real, se desconoce la eficiencia de ésta y las limitaciones que puede presentar. Una de las problemáticas que puede influir a la existencia de varias limitaciones es la falta de guía técnica que se ha tenido por parte del personal participante del proyecto PyrenEOS. Siendo otro de los factores, la falta de medios óptimos en el puesto de trabajo, teniendo el estudiante que aportar sus propios medios en el entorno domiciliario.

Aun así, a fin de comprobar el correcto funcionamiento, se pone en marcha la instalación y ejecución de la plataforma en un entorno simulado mediante máquinas virtuales como se representaba en la Figura 17, emulando la arquitectura de la Figura 28.

4.5.2. Resultados

Con el lanzamiento del programa en el entorno de evaluación del apartado 4.2.5. para los productos Sentinel-2, se obtuvieron que la mayoría de las descargas procedían de los repositorios CODEDE, SCIHUB y PEPS, sin embargo este último presentó a su vez más errores por tiempo excedido. Esto significa que se insertó como variable una mala estimación del tiempo máximo permitido por lo que se debería volver a plantear dicho estudio.

Un factor que ralentiza el proceso, es la descompresión del fichero.zip y su correspondiente cambio al directorio escogido, teniendo una demora por un archivo de media 800 MB de aproximadamente 50 segundos. Esto se puede traducir en dos problemas: la CPU necesita ser más potente o la estructura del código debe estar mejor organizada evitando pilas de funciones.

A modo demostrativo, se muestran las siguientes capturas con un ejemplo de aquellos procesos que no se han podido visualizar en la explicación del desarrollo del software.

Tras la inserción de parámetros a través del interfaz, y con `check_db.py` ejecutándose cada 10 minutos, éste último obtiene el proceso de descarga a seguir para los productos Sentinel-2 según las variables.

```

launch_S2_download.bat: Bloc de notas
Archivo Edición Formato Ver Ayuda

AINHOA\Universidad\TFG\DSERVER\Downloaders\do_codede_download.bat" "S2L1C" "2017-06-14" "2017-06-28" "-42.389" "-42.321" "68.379" "68.423" "58" "192.168.56.1"
AINHOA\Universidad\TFG\DSERVER\Downloaders\do_austrianm_download.bat" "S2L1C" "2017-06-14" "2017-06-28" "-42.389" "-42.321" "68.379" "68.423" "56" "192.168.56.1"
AINHOA\Universidad\TFG\DSERVER\Downloaders\do_peps_download.bat" "S2L1C" "2017-06-14" "2017-06-28" "-42.389" "-42.321" "68.379" "68.423" "80" "192.168.56.1"
AINHOA\Universidad\TFG\DSERVER\Downloaders\do_hnsdms_download.bat" "S2L1C" "2017-06-14" "2017-06-28" "-42.389" "-42.321" "68.379" "68.423" "100" "192.168.56.1"
AINHOA\Universidad\TFG\DSERVER\Downloaders\do_scihub_download.bat" "S2L1C" "2017-06-14" "2017-06-28" "-42.389" "-42.321" "68.379" "68.423" "150" "192.168.56.1"
AINHOA\Universidad\TFG\DSERVER\Downloaders\do_theia_download.bat" "S2L2A" "2017-06-14" "2017-06-28" "-42.389" "-42.321" "68.379" "68.423" "200" "192.168.56.1"
AINHOA\Universidad\TFG\DSERVER\Downloaders\do_scihub_download.bat" "S2L2A" "2017-06-14" "2017-06-28" "-42.389" "-42.321" "68.379" "68.423" "100" "192.168.56.1"
AINHOA\Universidad\TFG\DSERVER\Downloaders\do_unzip_and_stored.bat" "C/YYYY/MM/DD" "D:\SENTINEL" "C:\SENTINEL" "S2" "192.168.56.1"

```

Figura 40: Ejemplo del contenido del fichero autoprogramable para la descarga de productos Sentinel 2

En este, se puede apreciar como se establece correctamente el orden secuencial de la descarga desde los distintos repositorios. Además, el script check_db, genera la tarea de forma automática correctamente:



 sentinel_1_download	Listo	A las 8:00 todos los días
 sentinel_2_download	Listo	A las 6:00 todos los días

Figura 41: Tarea programada de la descarga desde check_db.py

Una vez que se descarga correctamente, el programa creará en la unidad de almacenamiento indicada el directorio de carpetas donde se guardará el producto sin errores.

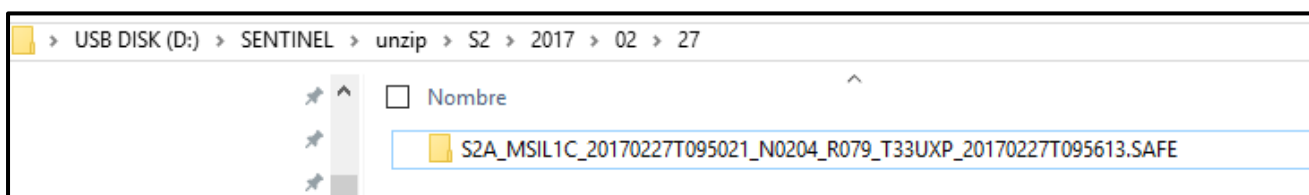


Figura 42: Ejemplo del almacenamiento de un producto tras su correcta descarga

Debido a la gestión del tiempo por parte de la autora, no se ha podido poner en funcionamiento la plataforma durante un periodo largo de días, lo cual habría contribuido al estudio del comportamiento de la plataforma y su correspondiente conclusión acerca de su eficacia.

5. CONCLUSIONES

Los objetivos principales del presente Trabajo Fin de Grado se dividían en dos tipologías: teóricos y prácticos. Los primeros consistían en la adquisición de conocimientos básicos sobre la teledetección espacial, Copernicus y el proyecto PyrenEOS. Éstos se consideran más que cumplidos, teniendo, a nivel personal, un gran aporte para el estudiante ya que se partía sin nociones básicas algunas.

En cuanto a la parte práctica, se estableció la descarga de imágenes Sentinel de manera automática sin ser una solución a nivel de usuario y contando con un interfaz de configuración y monitorización. Como se ha podido ver a lo largo del documento, este objetivo se ha cumplido notablemente, aportando además, una plataforma multiservidora gracias al estudio de investigación previo realizado por el estudiante con medios propios.

Sin embargo, atendiendo al título del documento *“Adecuación y Puesta en Producción de la Descarga Automática de Imágenes Sentinel en el Entorno Tecnológico de Gobierno de Navarra”*, se concluye la incoherencia entre los resultados finales y el objetivo de adecuar y poner en producción la herramienta creada. No obstante, esta problemática es debida a la falta de gestión en el proyecto PyrenEOS, hecho ajeno al estudiante en cuestión. A modo compensatorio, se ha intentado dotar de la máxima flexibilidad posible, dentro de las limitaciones en conocimientos, para la adecuación en distintas infraestructuras no solo de Gobierno de Navarra.

Con el desarrollo de la herramienta se han adquirido nuevos conocimientos de manera autónoma, relacionados técnicamente con el Grado cursado. Ejemplo de ello son los lenguajes de programación (sobre todo Python), automatización de tareas, funciones de archivos de procesamiento por lotes, esquemas de estructuración de software, uso de base de datos PostgreSQL, etc.

Haciendo autocrítica del trabajo realizado, se presentan posibles líneas de mejora a priori, y una vez puesta realmente en producción, comprobando su eficiencia, se obtendrán líneas de mejora a posteriori.

- Plataforma operativa en varios sistemas: actualmente sólo opera en Windows
- Carga de datos a PostGIS mediante ogr2ogr [44]: se trata de una librería más eficaz y rápida que la implementada para esta función.
- Notificación al usuario en caso de error: la herramienta muestra el asunto sólo cuando el usuario accede a la web
- Mayor flexibilidad en el código ante la aparición de nuevos repositorios
- Plataforma multiservidora según la zona geográfica: aprovechando la disponibilidad de varios servidores y debido a las limitaciones de resultados de estos, se podría implementar en un futuro un algoritmo donde según el tamaño del área elegido se reparta la descarga de todas las imágenes entre servidores de forma paralela, disminuyendo el tiempo de descarga.
- Realizar el proceso de pre-procesado y almacenamiento en una máquina más rápida o bien mejorar la estructura del código para ahorrar tiempo.



- Implementar el proceso de una primera carga inicial, pudiendo tener hasta 2 años de imágenes acumuladas.
- Configurar una segunda hora programada de descarga, en caso de tener fallo en las peticiones a todos los servidores debido, por ejemplo, a una posible avería implicando a la conexión.
- Añadir imágenes del satélite Landsat8 a fin de aprovechar posibles aplicaciones: se ha comentado que en un futuro muchas herramientas software se diseñarán con la combinación de este satélite y los Sentinel

En resumen, el software desarrollado cumple la gran mayoría de objetivos establecidos, lo cual personalmente resulta satisfactorio debido a los escasos concomimientos iniciales. Sin embargo, es una herramienta que presenta múltiples vías de mejora, sin haber tenido la oportunidad de su puesta en producción en un entorno tecnológico real.



BIBLIOGRAFÍA

- [0] “Programa operativo Interreg Poctefa”, *Poctefa*. [En línea]. Disponible en: <https://www.poctefa.eu/poctefa0713/arbol/index.jsp?id=d6c275d5ebd1c3d360465fe70c1e5b7a>
- [1] “Percepción remota”, *Centro de Investigación en Geografía y Geomática*. [En línea]. Disponible en: <http://www.centrogeo.org.mx/index.php/investigacion/lineas-de-investigacion/percepcion-remota>
- [2] “Imágenes de satélite”, *Instituto Nacional de Estadística y Geografía*. [En línea]. Disponible en: <http://www.inegi.org.mx/geo/contenidos/imgpercepcion/imgsatelite/elementos.aspx>
- [3] “Espectro electromagnético”, *Wikipedia*. [En línea]. Disponible en: https://es.wikipedia.org/wiki/Espectro_electromagn%C3%A9tico
- [4] “Tutorial sobre la percepción remota”, *Teledet*. [En línea]. Disponible en: <http://www.teledet.com.uy/tutorial-imagenes-satelitales/imagenes-satelitales-tutorial.htm>
- [5] “Fundamentos físicos de la teledetección”, *Universidad de Murcia*. [En línea]. Disponible en: <http://www.um.es/geograf/sig/teledet/fundamento.html>
- [6] “Basics of remote sensing”, *Grind GIS-GIS*. [En línea]. Disponible en: <http://grindgis.com/what-is-remote-sensing/know-basics-of-remote-sensing>
- [7] “Introduction to categorisation of objects from their data”, *Science Education through Earth Observation for High Schools*. [En línea]. Disponible en: <http://www.seos-project.eu/modules/classification/classification-c00-p05.html>
- [8] “Introduction to remote sensing of biomass”, *Intech open science*. [En línea]. Disponible en: <https://www.intechopen.com/books/biomass-and-remote-sensing-of-biomass/introduction-to-remote-sensing-of-biomass>
- [9] “Copernicus, Europe’s eyes on Earth”, *Copernicus*. [En línea]. Disponible en: <http://www.copernicus.eu/main/overview>
- [10] “Copernicus folleto español”, *Copernicus*. [En línea]. Disponible en: <http://copernicus.eu/main/Brochure>
- [11] “Copernicus, observing the Earth”, *ESA*. [En línea]. Disponible en: http://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Overview4
- [12] “Sentinel satellites”, *Copernicus*. [En línea]. Disponible en: <http://copernicus.eu/main/sentinels>
- [13] “Sentinel satellites”, *Copernicus*. [En línea]. Disponible en: <http://www.eumetsat.int/website/home/Satellites/FutureSatellites/CopernicusSatellites/index.html>
- [14] “Sentinel-1 SAR, Product Types and Processing Levels”, *ESA*. [En línea]. Disponible en: <https://sentinels.copernicus.eu/web/sentinel/user-guides/sentinel-1-sar/product-types-processing-levels>



- [15] "Sentinel-2, Processing Levels", *ESA*. [En línea]. Disponible en: <https://sentinels.copernicus.eu/web/sentinel/user-guides/sentinel-2-msi/processing-levels>
- [16] "Sentinel-3, Data products", *ESA*. [En línea]. Disponible en: <https://sentinels.copernicus.eu/web/sentinel/missions/sentinel-3/data-products>
- [17] "Presentación del proyecto transfronterizo PyrenEOS", *Gobierno de Navarra - noticias*. [En línea]. Disponible en: https://www.navarra.es/home_es/Actualidad/Sala+de+prensa/Noticias/2016/10/21/presentacion+proyecto+transfronterizo+PyrenEOS+Navarra+Copernicus.htm
- [18] "Borrador PyrenEOS_ES.pdf", *Autor: Socios del proyecto PyrenEOS e Interreg Poctefa* [No disponible en línea]
- [19] "Álvaro Huarte, SIG Developer", *Slideshare*. [En línea]. Disponible en: <https://www.slideshare.net/AlvaroHuarte>
- [20] "PostGIS – Spatial and Geographic Objects for PostgreSQL", *Postgis*. [En línea]. Disponible en: <http://postgis.net/>
- [21] "What is Geoserver?", *GeoServer*. [En línea]. Disponible en: <http://geoserver.org/about/>
- [22] "Copernicus Open Access Hub", *Copernicus Open Access Hub*. [En línea]. Disponible en: <https://scihub.copernicus.eu/>
- [23] "Copernicus Online Data Access", *EUMETSAT*. [En línea]. Disponible en: <https://coda.eumetsat.int>
- [24] "All the Copernicus Open Access Hub News", *Copernicus Open Access Hub*. [En línea]. Disponible en: <https://scihub.copernicus.eu/news/>
- [25] "Collaborative Ground Segment", *ESA*. [En línea]. Disponible en: <https://sentinels.copernicus.eu/web/sentinel/missions/collaborative>
- [26] "USGS Science for a changing world", *USGS (Servicio Geológico de los Estados Unidos)*. [En línea]. Disponible en: <https://www.usgs.gov/>
- [27] "Sentinel – Hub Browser", *Sentinel-Hub*. [En línea]. Disponible en: <http://www.sentinel-hub.com/>
- [28] "CEDA opensearch", *CEDA*. [En línea]. Disponible en: https://github.com/cedadev/ceda_opensearch
- [29] "Centre for Enviromental Data Analysis", *CEDA*. [En línea]. Disponible en: <http://data.ceda.ac.uk/>
- [30] "Sentinel2-L1C items assets catalog", *Planet Platform*. [En línea]. Disponible en: <https://www.planet.com/docs/reference/data-api/items-assets/sentinel2l1c/>
- [31] "Sentinel2 on Amazon Web Services", *AWS*. [En línea]. Disponible en: <https://aws.amazon.com/es/public-datasets/sentinel-2/> <http://sentinel-pds.s3-website.eu-central-1.amazonaws.com/>



- [32] “Google Cloud Platform – Sentinel 2”, *Google Cloud*. [En línea]. Disponible en: <https://cloud.google.com/storage/docs/public-datasets/sentinel-2>
- [33] “EO Cloud Platform - Earth Observation Innovative Platform Testbed Poland (EO IPT Poland)”, *EO CLOUD*. [En línea]. Disponible en: <http://www.cloudferro.com/en/eocloud/>
- [34] “peps_download.py”, *Oliver Hagolle*. [En línea]. Disponible en: https://github.com/olivierhagolle/peps_download
- [35] “Python 2.7. Release”, *Python*. [En línea]. Disponible en: <https://www.python.org/download/releases/2.7/>
- [36] “Everything about curl”, *curlhaxx*. [En línea]. Disponible en: <https://www.gitbook.com/book/bagder/everything-curl/details>
- [37] “Aria 2, Next generation download utility”, *aria2*. [En línea]. Disponible en: <https://aria2.github.io/>
- [38] “PostgreSQL: The world's most advanced open source database”, *PostgreSQL*. [En línea]. Disponible en: <https://www.postgresql.org/>
- [39] “XAMPP Installers and Downloaders”, *Apache Friends*. [En línea]. Disponible en: <https://www.apachefriends.org/es/index.html>
- [40] “Funciones de PostgreSQL”, *php.net*. [En línea]. Disponible en: <http://php.net/manual/es/ref.pgsq.php>
- [41] “Pyscopg: PostgreSQL + Python”, *psycopg*. [En línea]. Disponible en: <http://initd.org/psycopg/>
- [42] “Programming tasks with command line”, *Microsoft*. [En línea]. Disponible en: <https://technet.microsoft.com/en-us/library/bb490996.aspx>
- [43] “Sentinel 2 – Product Specification Document”, *ESA*. [En línea]. Disponible en: https://sentinel.esa.int/documents/247904/349490/S2_MSI_Product_Specification.pdf
- [44] “GDAL: ogr2ogr”, *GDAL*. [En línea]. Disponible en: www.gdal.org/ogr2ogr.html



ANEXO A: PROGRAMACIÓN DE TAREAS DE WINDOWS

El Programador de Tareas de Windows es una herramienta que ayuda a programar mediante interfaz gráfico o línea de comandos la automatización de tareas que realizan acciones cuando se produce un evento o a una hora concreta. En este apartado, se explicará cómo configurar tareas a través de la interfaz de Windows para el proyecto que nos ocupa, recordando que el directorio de los archivos no debe tener dependencias de usuario.

En primer lugar, se abrirá el Programador de Tareas (en Windows 10: Panel de Control > Sistema y Seguridad > Herramientas Administrativas) y se seleccionará la opción de la columna derecha “*Crear tarea básica*”. Aunque es posible escoger por igual “*Crear tarea*”, la elección anterior es más intuitiva ya que se tiene ayuda del asistente. La pantalla inicial instará a escribir el nombre de dicha tarea, procurando seleccionar uno intuitivo, y, de manera opcional, una descripción de esta.

El siguiente paso será escoger el desencadenador de esta, es decir, cuándo se va a iniciar esa tarea, cuál será el evento o el tiempo en concreto que va a detonar ese inicio. Como refleja la Figura A.1, existen diversos desencadenantes. Para nuestro caso, se seleccionará *Diariamente* y se indicará la fecha y hora en concreto. Hay que tener en cuenta que a pesar de configurar la tarea cada X minutos/horas/días, hasta que no se dé por primera vez esta fecha-hora no se repetirá la acción. Es por esto que es recomendable elegir una fecha-hora futura próxima.

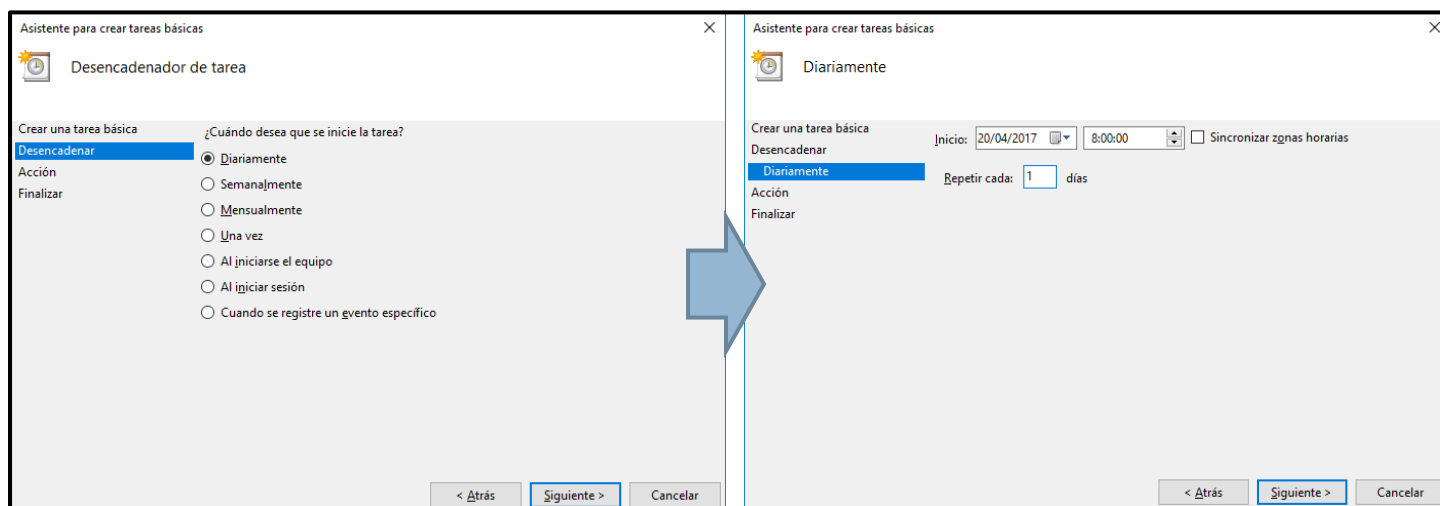


Figura A. 1: Selección de desencadenador de tarea

A continuación, se debe indicar la acción a realizar, es decir qué programa se va a ejecutar. Escogeremos *Iniciar un programa*, seguidamente se examinará el programa en cuestión, en este caso, el .bat de interés de la carpeta Web_Repositories_Test. No se requiere de argumentos adicionales.

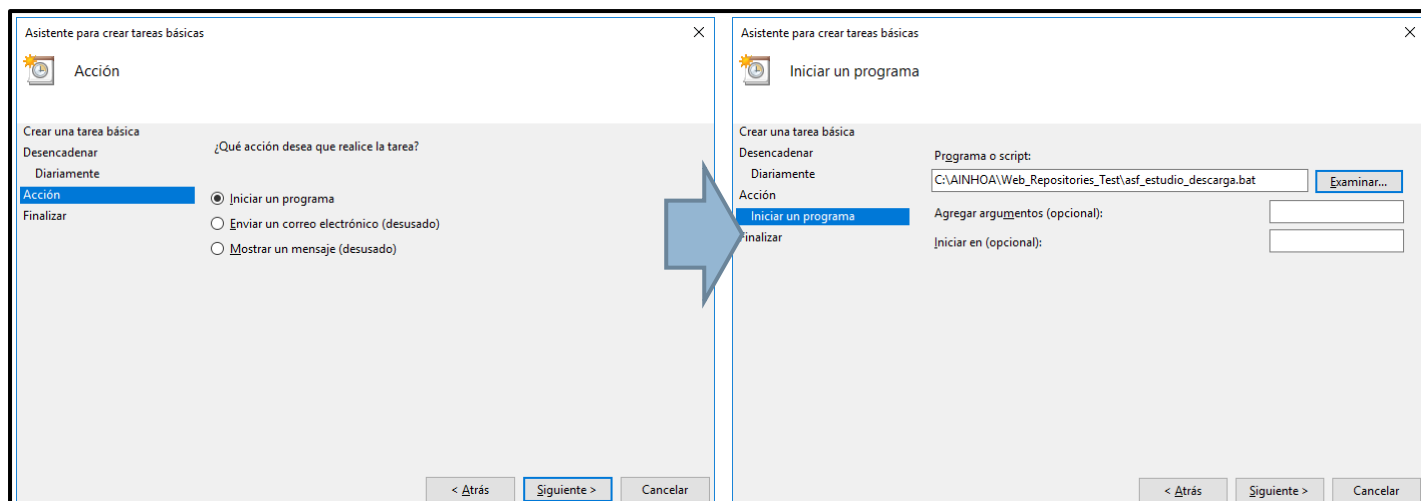


Figura A. 2: Selección de la acción de tarea

Terminando con el asistente de tareas, aparecerá un resumen de la configuración. De estar todo correcto, se seleccionará la opción *Abrir el diálogo Propiedades para esta tarea al hacer clic en Finalizar* y pulsaremos *Finalizar*. Con esto, saldrá el cuadro de Propiedades concretas para esta tarea. En la pestaña General se deberá seleccionar *Ejecutar solo cuando el usuario haya iniciado sesión*, teniendo en consideración que el “usuario” es la cuenta de administrador sin contraseña alguna. Además se escogerá la opción *Ejecutar con los privilegios más altos* para la lectura, escrita y ejecución de ficheros.

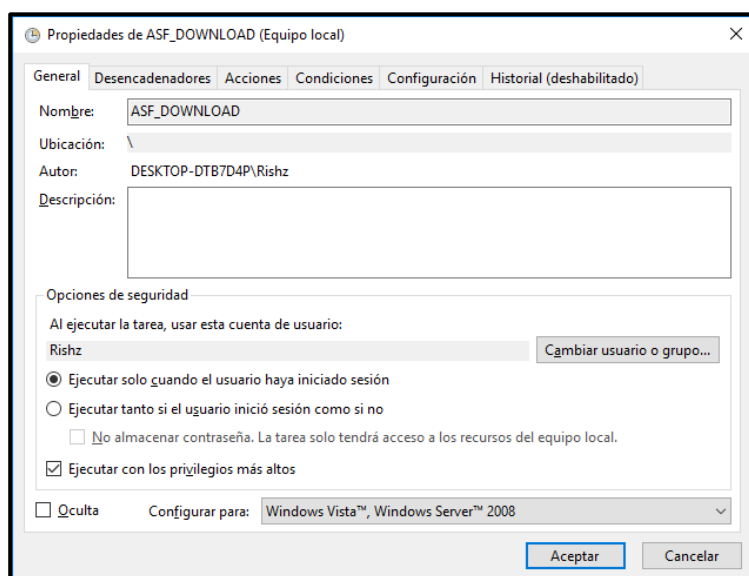


Figura A. 3: Opciones de seguridad de la tarea

Ahora mismo, solamente se tiene una tarea que se ejecutará una vez. Nuestro objetivo es la repetición de dicha acción en el periodo de tiempo deseado. Para ello, en la pestaña *Desencadenadores*, escogeremos la condición de interés y pulsaremos sobre *Editar*. La ventana emergente será la correspondiente con la Figura A.4. Habrá que seleccionar la opción *Repetir cada* y escoger entre las opciones la que más se ajuste a la que se quiere. También es importante cambiar la configuración por defecto de la opción *durante* y seleccionar de entre las dadas la que más convenga, en nuestro caso Indefinidamente. Para parar esta tarea se podrá eliminar o deshabilitar en el cuadro inicial del

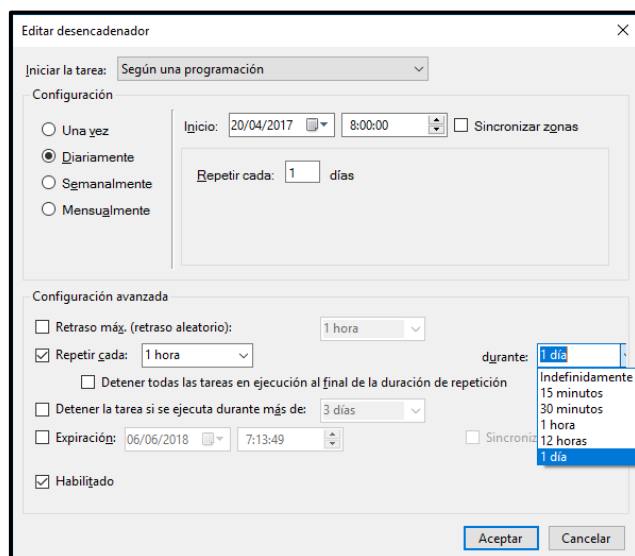


Figura A. 4: Configuración avanzada del desencadenador



Programador de Tareas. Es importante que en esta ventana este activada la opción **Habilitado**.

Por último, se deshabilitarán las opciones *Iniciar la tarea solo si el equipo está conectado a la corriente alterna* en **Condiciones**, de forma que si existe un corte de la corriente y el equipo pasa a usar la batería de manera provisional la tarea no se detenga y *Detener la tarea si se ejecuta durante de en Configuración*. Teniendo esto claro, se pulsará **Aceptar** y habrá finalizado la programación de la tarea

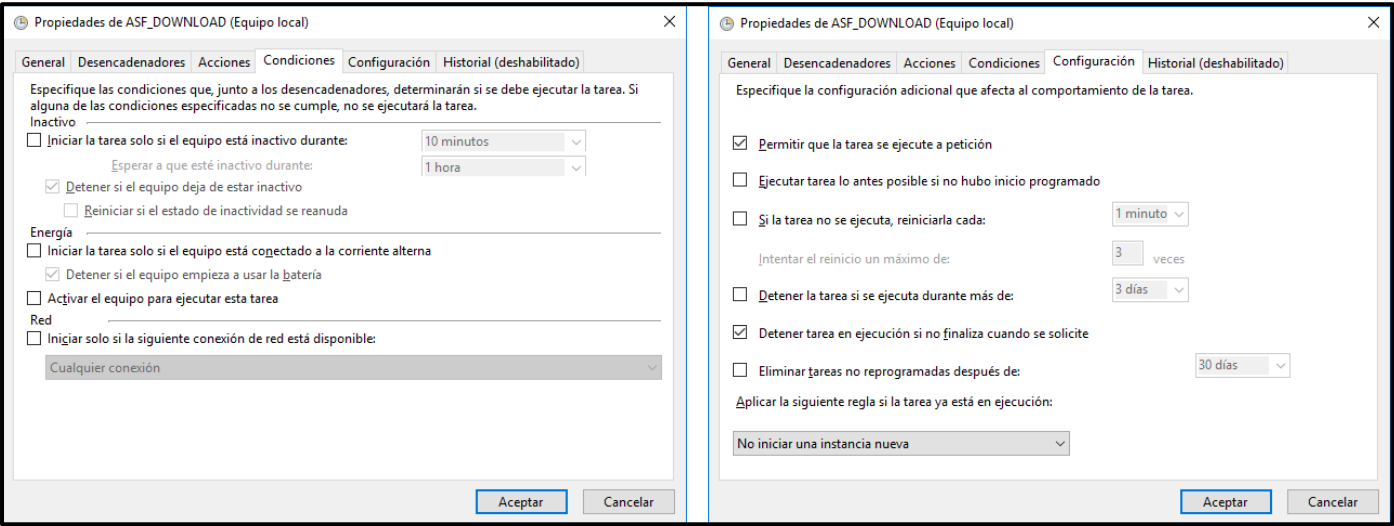


Figura A. 5: Condiciones y configuración adicional de la tarea



ANEXO B: MANUAL DE INSTALACIÓN EN EL SERVIDOR DE DESCARGA

El presente anexo contiene las indicaciones necesarias para el correcto funcionamiento del servicio de descarga en un sistema operativo Windows, siendo importante seguir dichas instrucciones tal cual se exponen.

PASO 1: Copia del programa

En el zip descarga TFG existen 2 carpetas: `DOWNLOAD_SERVER` y `LOCAL_WEB_SERVER`. La primera labor debe ser la copia del primero en la máquina destinada en el sistema a albergar el proceso de descarga.

Si bien la elección del directorio donde estará la carpeta es libre, es cierto que se debe tener atención a la hora de escoger este, teniendo que ser lo más corto posible debido al límite de caracteres en la ruta de archivos en Windows, `MAX_PATH`. Una segunda restricción es el requisito que el directorio no debe estar en dependencias de un único usuario y que sea una carpeta donde se dispongan de privilegios de lectura, escritura y ejecución.

PASO 2: Extracción de los archivos

Debido al tamaño total del contenido y con el fin de hacer más manejable su portabilidad, existen diversos archivos .zip cuyos ficheros se tienen que extraer. El destino de estos debe ser la misma carpeta en la que se encuentra el zip, evitando la creación de una nueva, es decir, seleccionando la opción “*Extraer aquí*”. Estos en concreto son, nombrados por orden de aparición, `download_folders.zip`, `downloaders.zip` dentro de *Downloaders*, `study_tools.zip` en *Web_Repositories_Test*, `settings.zip` en *Enviroment_Tools* y `python27.zip` que aparecerá en cuanto se extraigan los ficheros del anterior.

Una vez descomprimidos todos los zip, se puede proceder a su eliminación.

PASO 3: Configuración manual de parámetros

Para adaptarse al entorno, se deberán cambiar dos configuraciones, situándose en la carpeta `Check_DB` existen dos archivos a editar:

- `checkdb.bat`: insertar la dirección IP de la base de datos PostgreSQL, si se encuentra en la misma máquina, escribir *localhost*. En caso contrario y desconocer dicha IP, lea la sección Configuración de PostgreSQL del Anexo C.
- `launch_check.vbs`: este archivo sirve para que el anterior se ejecute en segundo plano, sin aparecer la consola de comandos cada vez que se haga correr. Es por esto que se le debe indicar la ruta completa al fichero `checkdb.bat`

La configuración final quedaría, para el caso en concreto, de acuerdo con las Figuras B.1 y B.2.



```
REM =====WEB SERVER IP for DB =====
REM =====This IP must be updated manually=====

set WEB_SERVER_IP=192.168.56.1
set parent_directory="%ENV_PATH%"
```

Figura B. 1: Configuración de IP en Check_DB/checkdb.bat

```
Set WshShell = CreateObject("WScript.Shell")

WshShell.Run chr(34) & "C:\Users\Ainhoa\TFG\DSERVER\Check_DB\checkdb.bat" & Chr(34), 0

Set WshShell = Nothing
```

Figura B. 2: Configuración de ruta completa en Check_DB/launch_check.vbs

PASO 4: Programar tarea de actualización de datos

Siguiendo las indicaciones del ANEXO A, se debe programar una tarea de Windows cuya acción será iniciar el programa launch_check.vbs. El desencadenador se recomienda que sea lo más inmediato, teniendo un tiempo de repetición de entre 5 y 15 minutos, es a libre elección pero se tiene que pensar que la función de este archivo es de actualizar los archivos de descarga según haya cambios en la base de datos. Hay que acordarse de seleccionar la opción *Habilitado* o la tarea no se ejecutará.

ANEXO C: MANUAL DE INSTALACIÓN EN EL SERVIDOR WEB

El presente anexo contiene las indicaciones necesarias para el correcto funcionamiento del servicio web y el servicio de base datos, centrado en un sistema operativo Windows, siendo importante seguir dichas instrucciones tal cual se exponen. Para empezar, se debe copiar la carpeta LOCAL_WEB_SERVER, la elección de la ruta a esta es libre siempre y cuando no se encuentren en dependencias accesibles bajo credenciales de usuario. Una vez hecho, descomprimir el archivo que viene en su interior, “Extraer aquí”, de forma que aparecerán dos directorios Settings y Web_Files. En este manual se utilizarán exclusivamente los archivos presentes en el primero, ver Figura C.1, teniendo que dejar intacto el contenido de Web_Files.

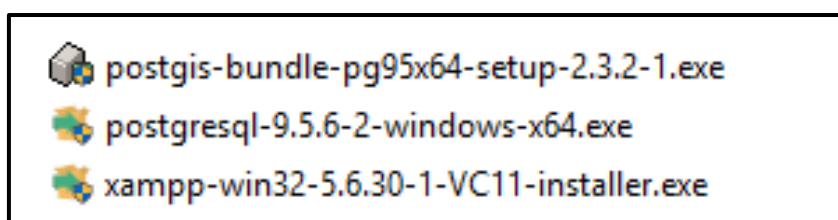


Figura C. 1: Contenido de la carpeta Settings dentro de LOCAL_WEB_SERVER

A partir de las versiones de la ilustración, se puede escoger otra versión más actualizada siempre y cuando PostGIS y PostgreSQL tengan la misma o serían incompatibles. Con esto, se comenzará con la instalación dividiendo el manual en dos apartados.

C-1: INSTALACIÓN DE XAMPP

Las indicaciones en esta sección son necesarias en el caso de que no se disponga de un servidor web local.

PASO 1: Ejecución del instalador de XAMPP

De acuerdo con la Figura C.1, se debe ejecutar xampp-win32-5.6.30-1-VC11-installer.exe con el fin de comenzar. Se puede dar el caso que aparezca un mensaje de advertencia nada más iniciar la instalación como ilustra la Figura C.2.

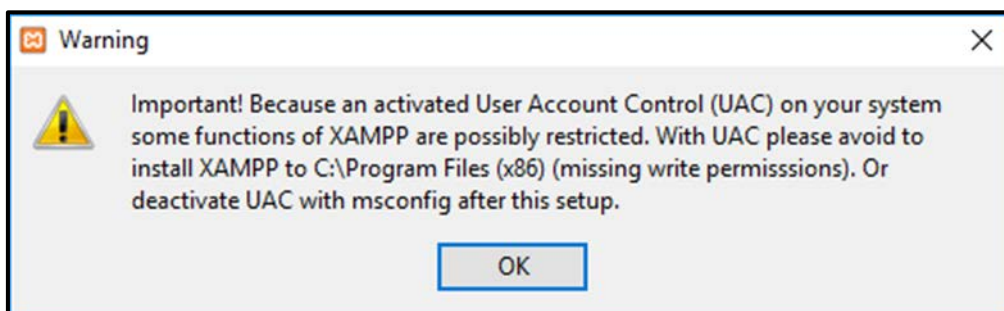


Figura C. 2: Aviso inicial de XAMPP por cuenta de usuario

Esto es debido a que la instalación se está realizando desde una cuenta de usuario, independientemente si posee privilegios de administrador o no. Se pulsará OK y se seguirán la recomendación del aviso, instalando el servidor en C:\xampp en lugar de C:\Program Files (x86).

Tras este mensaje, emergerá la ventana de bienvenida a la herramienta, teniendo que pulsar Next. A continuación, se tendrán que seleccionar los componentes deseados para el servidor, dejando la configuración por defecto, es decir, todos marcados como se muestra en la Figura C.3. El siguiente paso será escoger el directorio donde el servidor se instalará, recordando el aviso de la Figura C.2., es recomendable hacerlo en C:\xampp.

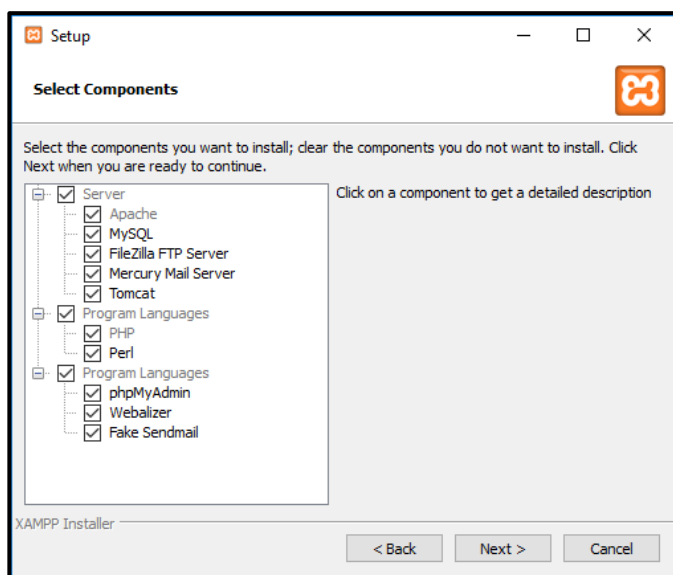


Figura C. 3: Selección componentes en XAMPP

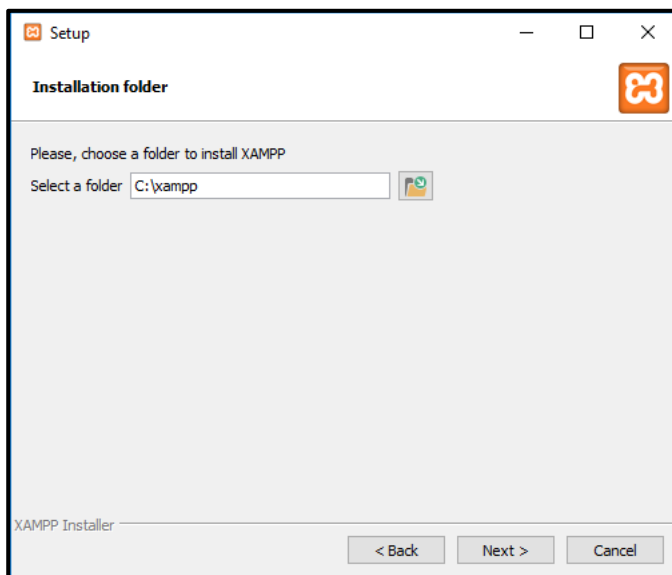


Figura C. 4: Carpeta de instalación de XAMPP

En las siguientes cuestiones se deberá desmarcar la opción “Learn more about Bitnami Xampp” y seleccionar Next para las dos pantallas siguientes. Se iniciará la instalación como tal y previo a su fin se deberá escoger las redes a las que se permite o se rechaza la conexión a XAMPP, es decir, se configurarán las reglas de entrada y salida para ApacheHTTPServer en el Firewall de Windows de la máquina.

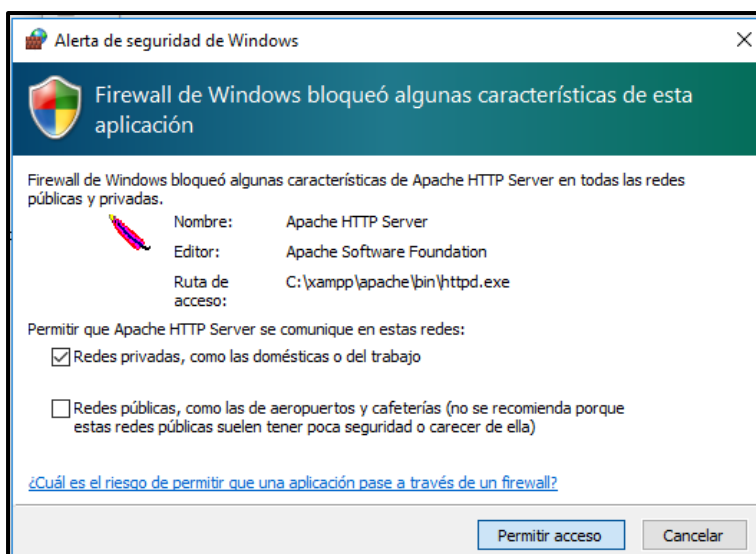


Figura C. 5: Mensaje firewall de Windows para XAMPP

La elección dependerá de la situación en cada caso, no siendo decisiva ya que estas reglas son modificables en la configuración avanzada del Firewall de Windows. Terminada la instalación, marcar la opción “Start Control Panel now” para continuar con el siguiente paso.

PASO 2: Puesta en marcha del servicio

Con el Panel de Control abierto, se pulsará sobre Start a los módulos Apache y MySQL con el fin de comprobar su correcto funcionamiento. Los puertos por defecto son para el primero el 80 (http) y 443(https) y 3306 para MySQL. Si alguno de estos se encuentran ocupados debido a otras aplicaciones instaladas que las usan, el Panel de Control informará de esta situación como se ejemplifica en la Figura C.6.

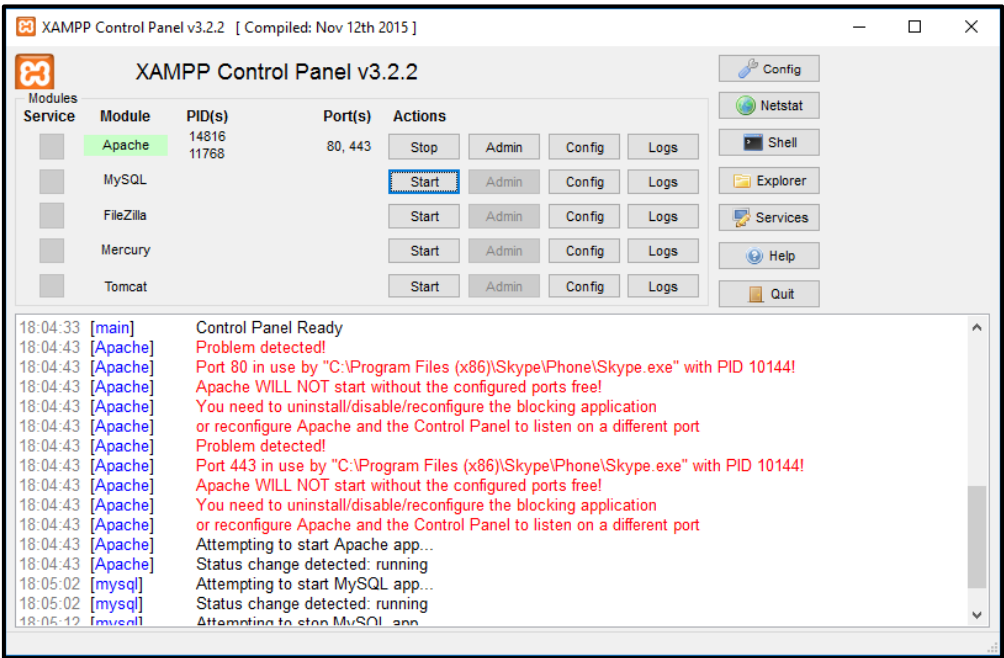


Figura C. 6: Mensaje panel de control en el caso de puertos ocupados

Para mayor comodidad a la hora de acceder a la URL del servidor, es recomendable cambiar la configuración de puertos de la aplicación que se esté usando siempre y cuando no sea de mayor importancia que el servidor web a instalar, en el caso de la Figura C.6, Skype.

En caso, de estar todo en orden a la hora de poner en marcha Apache y MySQL la información que proporcionará el panel de control será como la Figura C.7.

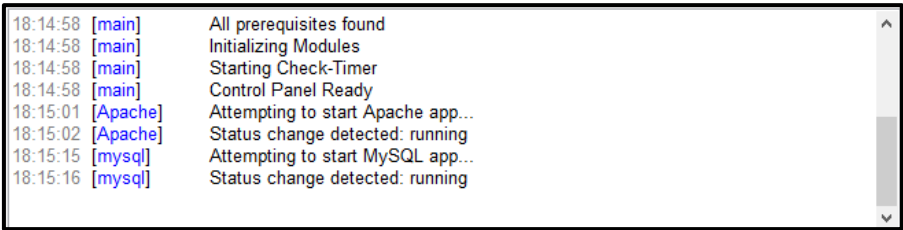


Figura C. 7: Mensaje panel de control iniciados Apache y MySQL



Sin embargo, al estar en una cuenta de usuario el servicio no está configurado para que arranque de forma automática cada vez que se inicia la máquina. Para solucionar esto, en C:\xampp, pulsar botón derecho sobre xamp-control.exe y clicar “Ejecutar como administrador”. Se abrirá de nuevo el Panel de control pero en este caso con la columna denominada Service activada como ilustra la Figura C.8. Si los servicios Apache y MySQL se encuentran corriendo, pararlos.

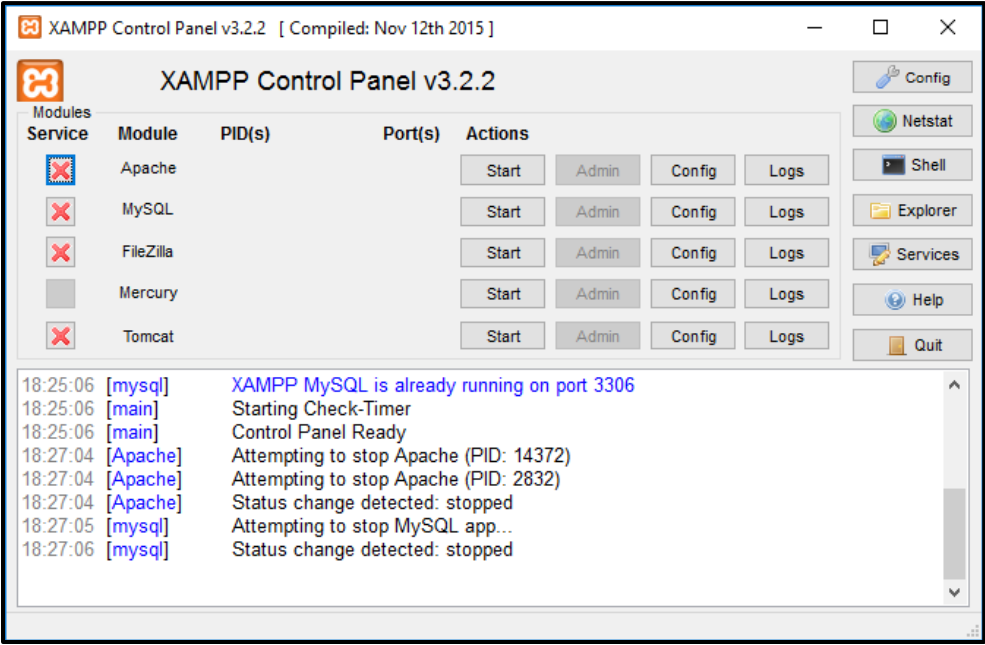


Figura C. 8: Panel de control como administrador

Pulsando sobre la X del servicio correspondiente a Apache, y contestando a Yes con la cuestión que se lanzará, el servicio quedará activado (tic verde). Realizar el mismo proceso para el módulo MySQL, y una vez hecho, poner en marcha ambos servicios (Start). Con este proceso, cada vez que la máquina donde está instalado XAMPP se inicia, arrancará automáticamente el servicio web.

A fin de comprobar su funcionamiento abrir el navegador y escribir <http://localhost> o <https://localhost>. Si se desea comprobarlo en un ordenador situado en la misma red, se deberá abrir la consola de comandos de Windows y escribir `ipconfig` para ver el listado de las direcciones IP de la máquina, buscando aquel adaptador que se encuentre en la red donde estará también el cliente, para el caso de la simulación, VB Host-Only Network.

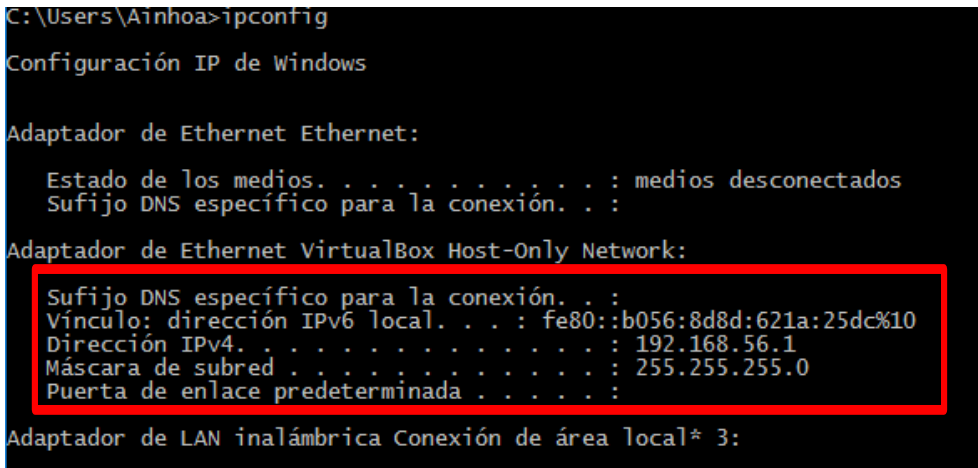


Figura C. 9: Dirección IP del servidor web



En otra máquina de esa red, tras comprobar que se puede realizar ping a la dirección del servidor, se tendrá que escribir en el navegador <http://IP> , saliendo la página de bienvenida de XAMPP. En caso de no poder hacer ping, se deberá al firewall de la máquina cliente y la solución será crear reglas de entrada y salida para esa dirección IP destino.

PASO 3: Carga de archivos de la página

Manteniéndose en el panel de control, el siguiente paso será pulsar sobre Config y abrir el archivo `httpd.conf`. Buscando `DocumentRoot` y `<Directory>` se deberá cambiar la configuración de ficheros web por defecto por la ruta donde se encuentra la carpeta `Web_Files` que se ha extraído al principio de este manual. Es decir, se cambia la ubicación de los archivos que conformaban la web por defecto, a los ficheros que en conjunto formarán la web de visualización y monitorización de PyrenEOS. Para navegar de forma segura, se debe proceder de igual forma en el archivo `http-ssl.conf` pero en este caso cambiando únicamente la ruta en `DocumentRoot`

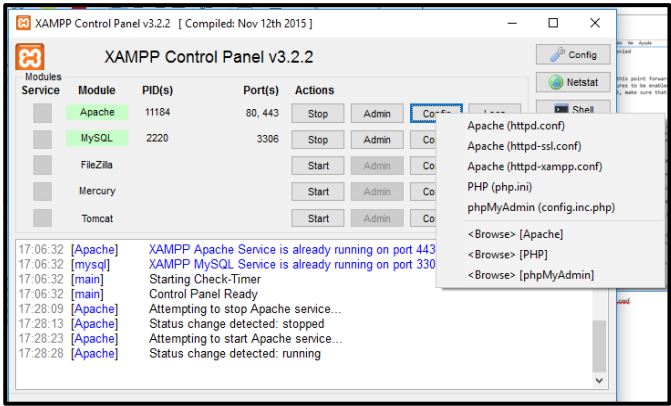


Figura C. 10: Acceso a archivos de configuración de XAMPP

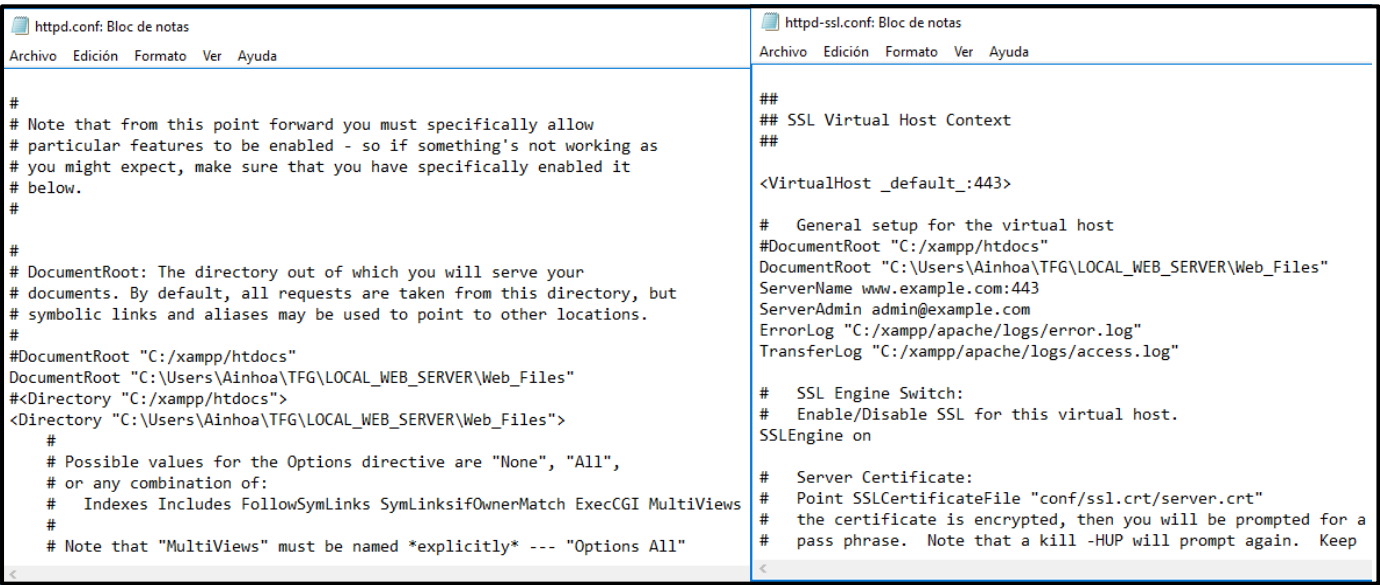


Figura C. 11: Cambio de ubicación de la página web en `httpd.conf` y `httpd-ssl.conf`

PASO 4: Extensión de PostgreSQL en PHP

Por último, para que el motor PHP del servicio web pueda conectarse a la base de datos, se deberá habilitar la extensión de PostgreSQL en este, permitiendo así el uso de las funciones específicas para realizar consultas y extraer información de la base de datos. Siguiendo la metodología del paso anterior, se abrirá el fichero `php.ini` y en la sección *Windows Extensions*, descomentar aquellas que contenga `php_pdo_pgsql.dll` y `php_pgsql.dll` como se muestra en la Figura C.12.



```
extension=php_pdo_mysql.dll
;extension=php_pdo_oci.dll
;extension=php_pdo_odbc.dll
extension=php_pdo_pgsql.dll
extension=php_pdo_sqlite.dll
extension=php_pgsql.dll
;extension=php_shmop.dll
```

Figura C. 12: Extensiones de Postgre a habilitar en php.ini

Reiniciar Apache y se tendrá un servidor web funcional para el proyecto PyrenEOS.

C-2: INSTALACIÓN DE POSTGRESQL

La puesta en marcha de esta base de datos será necesaria para el sistema diseñado, en caso de elegir otro tipo de sistema de gestión de datos, se tendrá que adaptar los archivos que contactan a la BBDD a la nueva elegida. Se proporciona a continuación las indicaciones de cómo proceder a la instalación de PostgreSQL.

PASO 1: Ejecución del instalador de PostgreSQL

Poniendo en marcha el archivo postgresql-9.5.6-2-windows-x64.exe, al comenzar con el proceso aparecerá una pantalla de bienvenida al instalador; seguidamente se preguntarán informaciones acerca de la configuración inicial. En primer lugar, se tendrá que escoger el directorio donde se instalará el programa, siendo recomendable la configuración por defecto que viene dada. A continuación, se deberá elegir la carpeta donde se almacenarán los datos, es decir, la información de esquemas, tablas, datos de tablas, etc. En este caso se ha dejado nuevamente la configuración por defecto. Sin embargo, en un entorno de producción los datos se suelen almacenar en un directorio distinto al de instalación, como siempre, teniendo que tener privilegios sobre este.

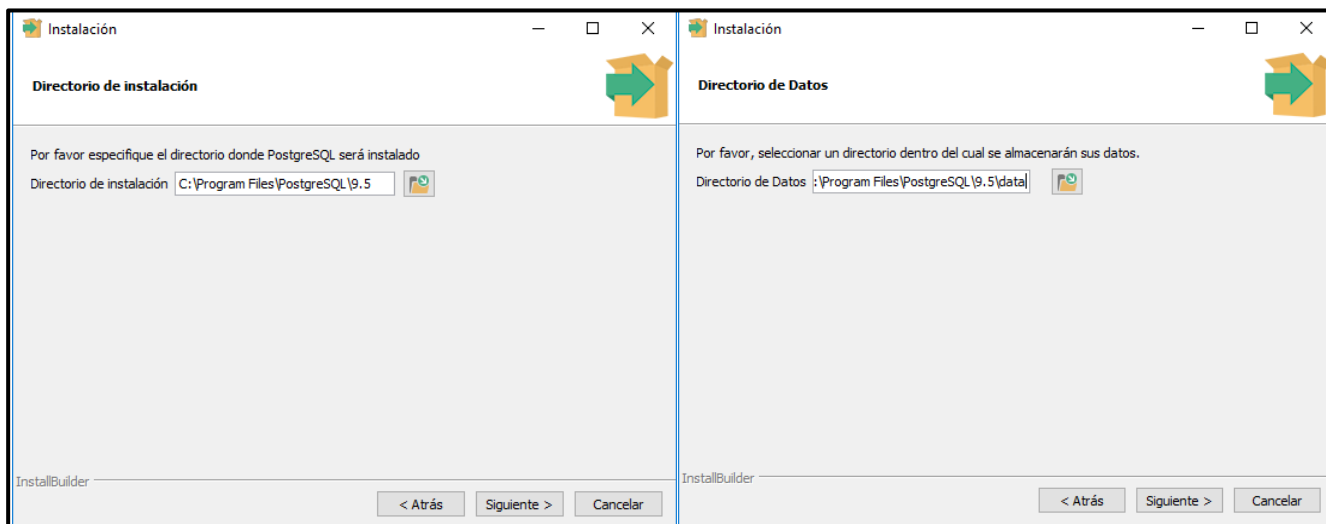
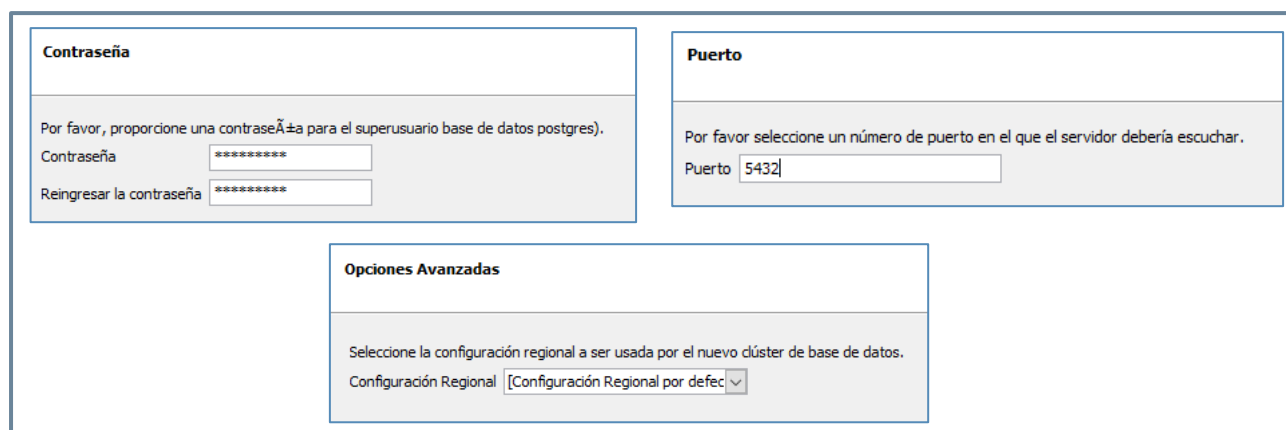


Figura C. 13: Directorio de instalación y de datos para PostgreSQL

Siguiendo con el proceso, se pedirá una contraseña para el superusuario postgres, este es aquel que tiene acceso a todas las bases de datos de PostgreSQL, es decir, el administrador del gestor de datos. Los últimos parámetros a establecer serán el puerto, dejando por defecto 5432 (el servidor web y el de descarga realizan la conexión a este puerto, por lo que si se cambia se deberían cambiar todos

los archivos que comuniquen con este) y la codificación que, de nuevo, se dejará la dada por el instalador.



Contraseña

Por favor, proporcione una contraseña para el superusuario base de datos postgres).

Contraseña

Reingresar la contraseña

Puerto

Por favor seleccione un número de puerto en el que el servidor debería escuchar.

Puerto

Opciones Avanzadas

Seleccione la configuración regional a ser usada por el nuevo clúster de base de datos.

Configuración Regional

Figura C. 14: Elección de contraseña, puerto y codificación para PostgreSQL

Cuando se notifique de la finalización de la instalación, antes de pulsar Terminar, desmarcar la opción Lanzar Stack Builder, ya que este listará todos los plugins disponibles para la BBD.

PASO 2: Instalación del plugin PostGIS

El siguiente complemento será necesario para que las bases de datos posean analítica geométrica, necesaria para la carga de datos de los ráster (matriz de píxeles) de una imagen satélite. Ejecutando `postgis-bundle-pg95x64-setup-2.3.2-1.exe`, se tendrá que aceptar como paso previo el acuerdo de licencia. Seguidamente el asistente preguntará por los componentes a instalar, como muestra la Figura C.15, teniendo que seleccionar en este caso PostGIS. La segunda opción, crear una base de datos espacial, no es un paquete o complemento como tal, sino que al terminar la instalación crearía una base de datos con funciones PostGIS, lo cual no interesa en este caso.

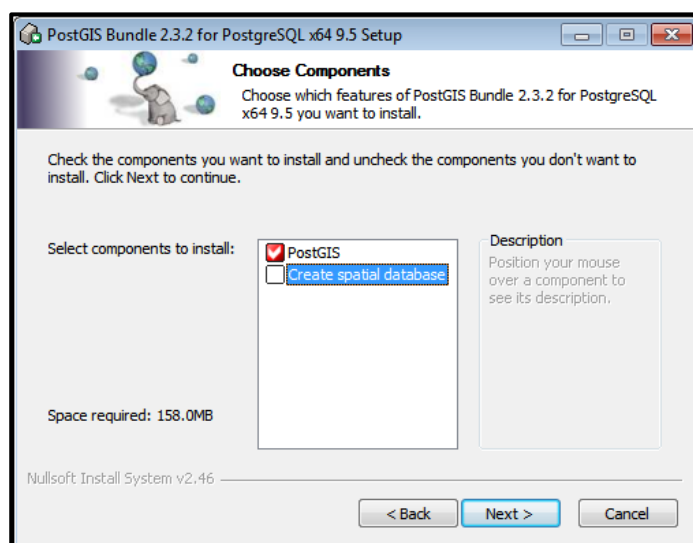


Figura C. 15: Elección complementos PostGIS

Al igual que ocurría en el paso anterior, se pedirá indicar el directorio donde PostGIS se instalará dejando por defecto el que se sugiere el instalador. A continuación, se deberán introducir usuario, contraseña y puerto a la base de datos PostgreSQL, es decir usuario: postgres, contraseña: la establecida en el paso anterior y puerto 5432. NOTA: Este apartado se omite en Windows 10, pero la instalación sigue siendo correcta.

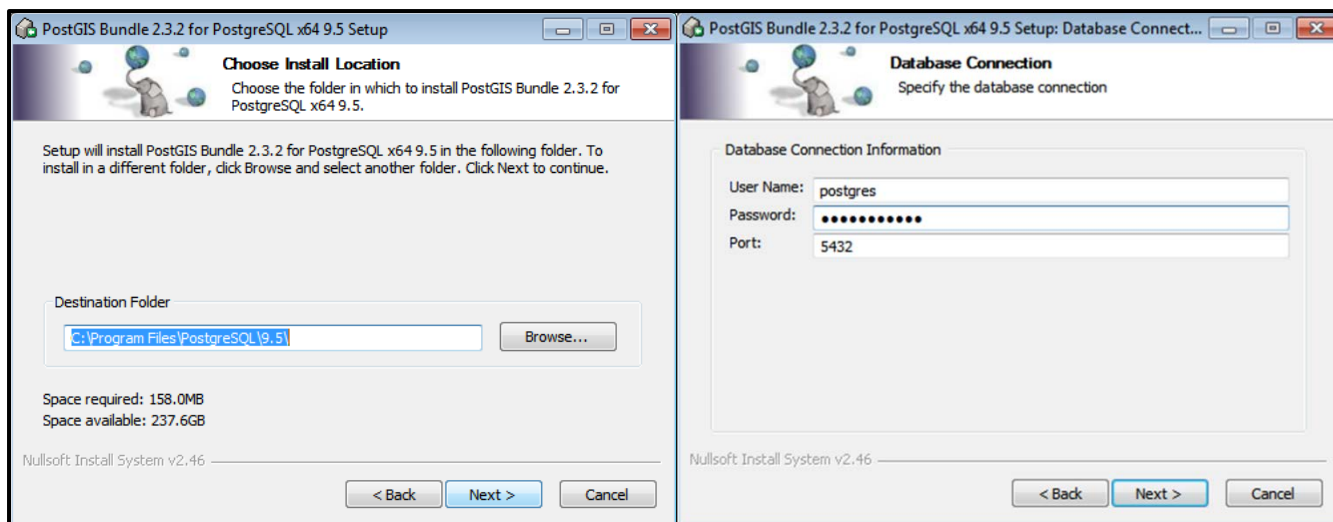


Figura C. 16: Directorio de instalación de PostGIS y parámetros para la conexión a la base de datos

Comenzará tras esto la instalación como tal, quedando pausado el proceso con la aparición de 3 cuestiones al usuario a las que se deberá contestar que sí con el fin de que PostGIS incluya variables de entorno GDAL (biblioteca que permite la lectura y escritura de datos en formato geoespacial) y sea capaz de manejar rásters. Tras este último punto, quedará instalado el plugin de PostGIS

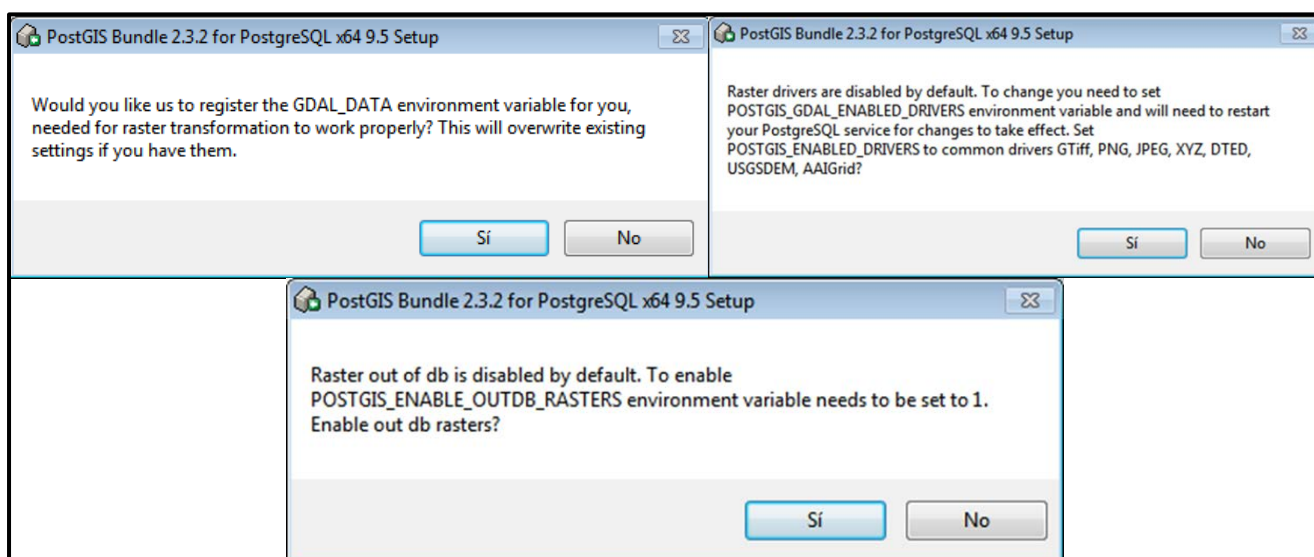


Figura C. 17: Aceptación variables de entorno GDAL

PASO 3: Preparación de la base de datos para la aplicación

Las instrucciones dadas en este paso tendrán como objetivo la configuración correcta adaptada al software desarrollado, minimizando lo máximo posible la configuración manual por parte del técnico. Con este fin, se deberá crear un rol de usuario propietario de las dos bases de datos a crear y una primera tabla con acceso para el administrador del servidor web.

Para empezar, se tendrá que abrir dentro de la carpeta PostgreSQL del paso 1, la aplicación gráfica *pgadmin*. El programa pedirá la contraseña necesaria para conectarse al servidor, que será la del

superusuario postgres que se estableció al instalar PostgreSQL en el paso 1. Para mayor comodidad en futuras aperturas de esta aplicación, se seleccionará la opción “Store password” de forma que no vuelva a surgir la pantalla correspondiente con la Figura C.18 cada vez que se quiera conectar al servidor.

Con el visor ya disponible, pinchar sobre PostgreSQL 9.5 en el menú de la derecha. Se desplegarán 4 opciones: Databases, Tablespaces, Group Roles y Login Roles.

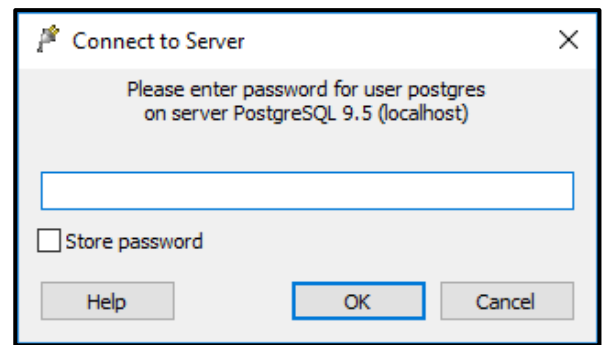


Figura C. 18: Contraseña para la conexión al servidor PostgreSQL

Antes de proceder a la creación de las bases de datos, se deberá forma un rol de login, esto es, la creación de credenciales de usuario que sean los administradores de unas bases de datos concretas pero no de PostgreSQL. Se puede plantear la posibilidad de utilizar el usuario postgres, sin embargo, hay que pensar que estas credenciales figurarán en los archivos PHP y los scripts del servidor de descarga, por lo que, por motivos de seguridad, es mejor la creación de un usuario.

Dicho esto, se creará el nuevo rol haciendo click derecho sobre Login Roles > New Login Role con lo que emergerá una pequeña ventana con los parámetros de configuración. Las credenciales DEBEN ser, de acuerdo con la aplicación desarrollada, usuario: **user_sentinel**, contraseña: **sentinel_pass** con privilegios de superusuario.

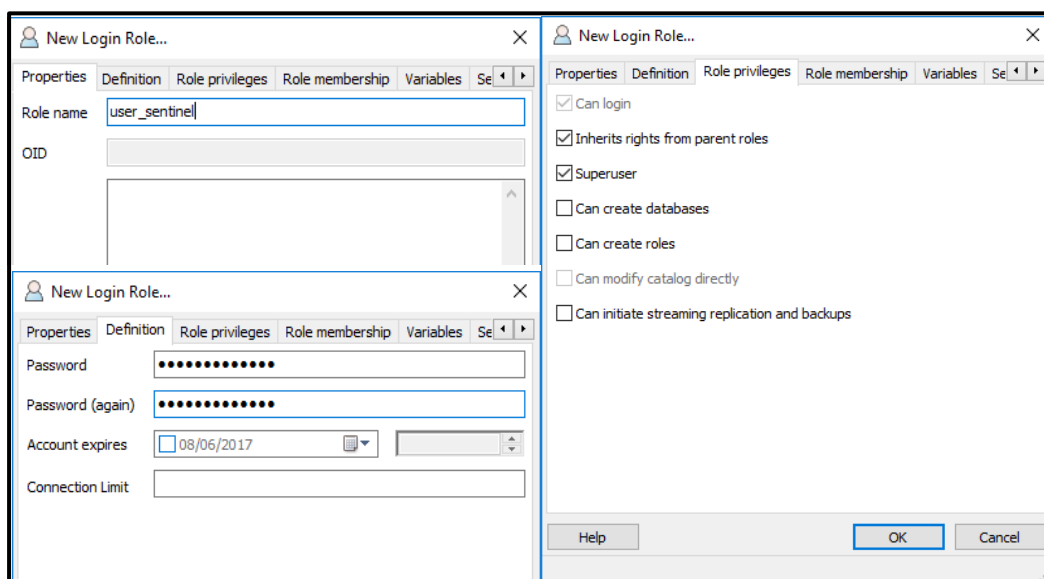


Figura C. 19: Configuración usuario sentinel

Una vez creado, presionando el botón derecho sobre Databases, se escogerá New Data Base, creando una base de datos de nombre **configuration_db** y con propietario (owner) el usuario **user_sentinel**. El mismo procedimiento se utilizará para la segunda base de datos, denominada **sentinel_data_db**. El motivo de la distinción en bases de datos y no en tablas, es que la segunda tendrá una extensión de PostsGIS, conteniendo tablas con metadatos de las imágenes y será la bbdd a la que GeoServer mandará peticiones; mientras que la base de datos de configuraciones contendrá tablas con datos propios de estado y configuración del sistema.

Para que **sentinel_data_db** posea analíticas geográficas, se deberá seleccionar esta y pulsar sobre la lupa “SQL” en la barra superior, con lo que aparecerá la siguiente pantalla:

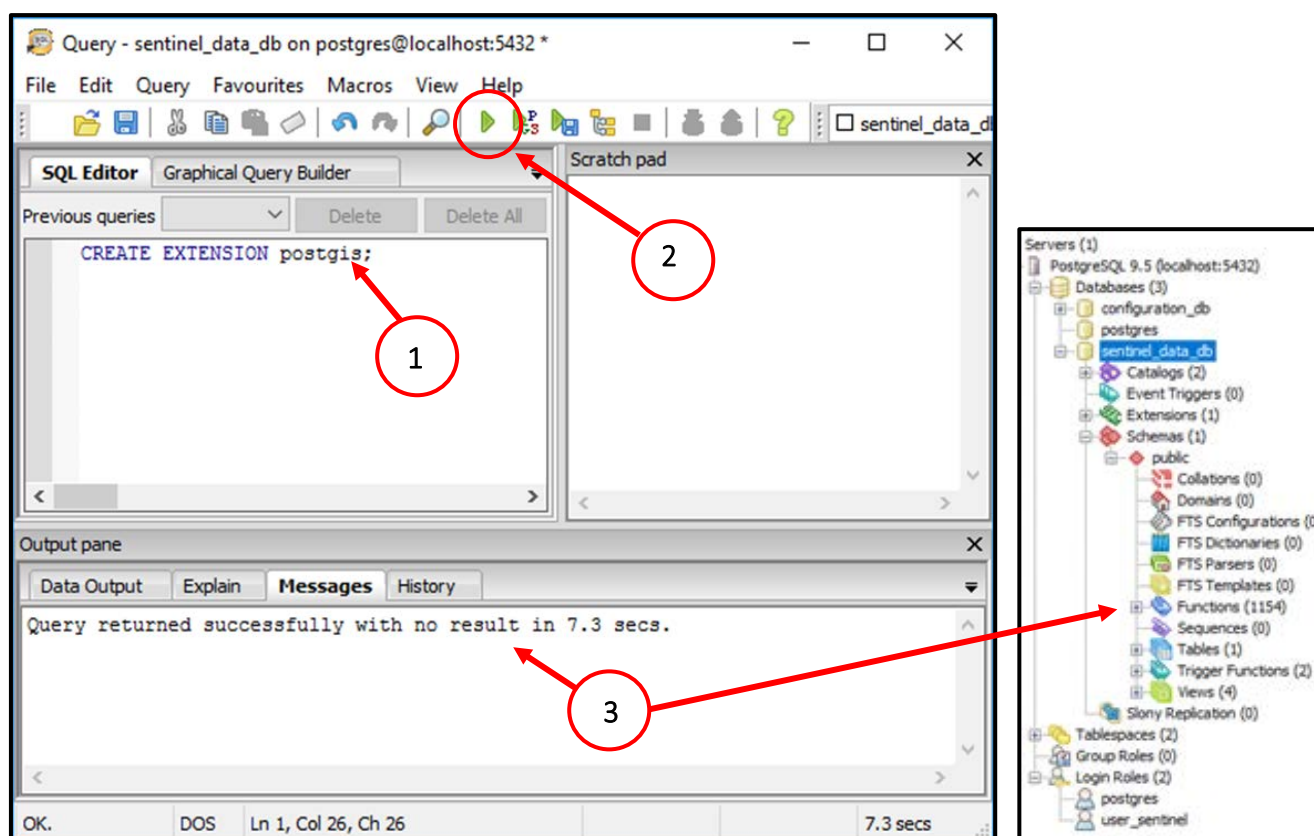


Figura C. 20: Añadir extensión PostGIS a sentinel_data_db

Siendo intuitivos los pasos de la Figura C.20, se deberá escribir en el editor `CREATE EXTENSION postgis;` y pulsar sobre ejecutar. Volviendo al menú inicial y desplegando la base de datos a la que se le añadido este complemento, se podrá ver que ahora existen 1154 funciones en ella.

El último punto a cumplir será la creación de una tabla dentro de **configuration_db** que contenga las credenciales del administrador de la aplicación web. Situándose en esta base de datos > Schemas > public > Tables y haciendo click derecho sobre este último se deberá seleccionar “New Table...”. El nombre de esta tiene que ser **info_account** y los datos por los que estará formada pertenecerán a las columnas **username**, **passwd**, **userrole**, **name** y **surname**. Para añadir estas últimas, se tiene que pinchar sobre la pestaña Columns, pulsar Add y escribir en el apartado Name el nombre de la columna y en Data Type el tipo de datos de estas. En este caso, las 5 serán de formato character varying (cadena de caracteres sin longitud definida).

Con todas las columnas añadidas, quedará definir la clave primaria de la tabla, para ello, se accederá a la pestaña Constraints y se desplegará el selector inferior eligiendo la opción Primary Key, pulsando a continuación sobre Add. El nombre que se le dará será **pk1** y la columna a la que se asignará esta clave va a ser username, de forma que no puede existir dos usuarios con el mismo nombre. Para realizar este último punto, en la pantalla que emerge al elegir la clave, se tendrá que escribir el nombre en la pestaña Properties y Columns seleccionar, username y pinchar sobre Add. Con todo este proceso

terminado (darle a OK) se creará la tabla deseada. A fin de comprobar su correcta implementación, se deja a continuación la Figura C.21, representando la declaración SQL para la formación de la tabla que aparecerá en el panel gráfico tras finalizar la asignación de variables de la tabla.

```
-- Table: public.info_account
-- DROP TABLE public.info_account;

CREATE TABLE public.info_account
(
    username character varying NOT NULL,
    passwd character varying,
    userrole character varying,
    name character varying,
    surname character varying,
    CONSTRAINT pk1 PRIMARY KEY (username)
)
WITH (
    OIDS=FALSE
);
ALTER TABLE public.info_account
OWNER TO user_sentinel;
```

Figura C. 21: Declaración SQL de la creación de info_account

Concluyendo este punto, se deberá añadir de forma manual los datos correspondientes a las credenciales del administrador. El role deberá ser de forma obligada “admin” y la contraseña insertada tiene que estar encriptado por el hash bcrypt. Ejemplificación de esto queda ilustrado en la Figura C.22.

username [PK] character varying	passwd character varying	userrole character varying	name character varying	surname character varying
ADMIN	\$2a\$04\$MGlHiLeowdTm0fwoI/KZSebfp2GZ/nwv/waaN1KQsxBLCsXBPwI'	admin		

Figura C. 22: Credenciales de administrado en info_account

Con todos los puntos correctamente seguidos en este paso, la creación de las tablas restantes no será problema para el usuario si no que los scripts se encargarán de forma automática de esto.

PASO 4: Acceso remoto a la base de datos

Hasta el momento sólo se dispone de un servidor PostgreSQL accesible desde la máquina local, sin embargo, en la arquitectura planteada en este proyecto, la base de datos debe ser accesible desde diferentes puntos de la misma red. En primer lugar se debe conocer la dirección IP del servidor en la LAN, para el presente caso será la misma que el servidor web: 192.168.56.1 y red 192.168.56.0/24. De ser distinta, se tendrá que consultar con ipconfig a través del terminal de Windows.

En la ruta donde se decidió el almacenamiento de los datos de PostgreSQL en el paso 1, existe el archivo pg_hba.conf en el que se deberá añadir una nueva línea con la que se permita la conexión de los equipos de esa subred, método que ilustra la Figura C.23.

```
# Put your actual configuration here
# -----
#
# If you want to allow non-local connections, you need to add more
# "host" records. In that case you will also need to make PostgreSQL
# listen on a non-local interface via the listen_addresses
# configuration parameter, or via the -i or -h command line switches.


# TYPE      DATABASE        USER            ADDRESS                 METHOD

# IPv4 local connections:
host        all             all             127.0.0.1/32            md5
host        all             all             192.168.56.0/24         trust
# IPv6 local connections:
```

Figura C. 23: Permitir conexión remota a PostgreSQL



Dependiendo de la versión de PostgreSQL, el técnico tendrá que cambiar la configuración de direcciones a las que se permite escuchar para establecer conexión. En la misma carpeta que en el párrafo anterior, se encuentra el fichero postgresql.conf, en el que se tendrá que comprobar que se permite la escucha de todas las direcciones IP, de no ser así, habrá que modificar esa línea como aparece en la Figura C.24

```
#-----  
# CONNECTIONS AND AUTHENTICATION  
#-----  
# - Connection Settings -  
  
listen_addresses = '*'
```

Figura C. 24: Permitir conexión remota en el archivo postgresql.conf

Para finalizar con la instalación completa, se deberá comprobar en el Firewall de la máquina que aloja el servidor PostgreSQL si se permite la entrada de tráfico TCP/IP con de destino el puerto 5432 y su correspondiente salida. En caso negativo, habrá que crear reglas para el Firewall que lo permitan. A continuación se exponen los pasos para ello, teniendo pocas restricciones en este caso al ser una red formada por máquinas virtuales. En el Firewall de Windows con seguridad avanzada y sobre reglas de entrada, seleccionar “Nueva Regla”, ídem para la de salida:

Regla de entrada:

- Tipo de regla: Puerto, controla las conexiones de un puerto TCP o UDP
- Protocolos y puertos: Aplicar sobre TCP a puertos locales específicos, en concreto, 5432
- Acción: Permitir la conexión
- Perfil: Aplicar la regla sobre todos los perfiles (dominio, privado y público)
- Nombre: PostgreSQL-input

Seleccionando la regla y presionando en el menú derecho la opción Propiedades.

- Ámbito: Cualquier dirección en Dirección IP local y Estas direcciones IP en Dirección IP remota, en concreto, 192.168.56.0/24

Regla de salida:

- Tipo de regla: Puerto, controla las conexiones de un puerto TCP o UDP
- Protocolos y puertos: Aplicar sobre TCP a todos los puertos remotos
- Acción: Permitir la conexión
- Perfil: Aplicar la regla sobre todos los perfiles (dominio, privado y público)
- Nombre: PostgreSQL-output

Seleccionando la regla y presionando en el menú derecho la opción Propiedades.

- Protocolos y puertos: Seleccionar en Puerto local la opción Puertos específicos e insertar 5432

Comprobando el acceso desde otra máquina a la base de datos, queda por concluida finalmente la instalación en la máquina servidor web + base de datos.

ANEXO D: MANUAL DE USO DEL PORTAL WEB

El presente anexo consiste en un pequeño tutorial acerca del uso del portal web, orientado especialmente a aquellos usuarios con permisos más allá de su visualización.

Tras ingresar correctamente con las credenciales de usuario, aparecerá una primera página principal de bienvenida y a la derecha un menú de opciones. Empezando de arriba y descendiendo en el menú, se explica a continuación el uso de cada sección.

PATH CONFIGURATION



The screenshot shows the 'SENTINEL PRODUCT INGESTOR' web interface. On the left is a vertical menu with the following items: 'Path Configuration' (highlighted), 'Download Configuration', 'Study Tools', 'Status', 'Add Users', and 'Exit'. The main content area is titled 'Path Configuration' and contains a form with three input fields: 'Path where zip files would be stored:' with the value 'C:\X\Y\Z...', 'Path where unzip files would be stored:' with the value 'D:\X\Y\Z...', and 'Folder schema in storage disks:' with a dropdown menu showing 'Collection/YYYY/MM/DD'. Below these fields are two buttons: 'SET' and 'RESET'. At the bottom of the form, there is a note: 'Please, be sure that there are permissions to write, read and execute files on these disks. Due to MAX_PATH limitation, try to choose short paths'.

Figura D. 1: Pantalla Path Configuration de la web Sentinel Product Ingestor

Desde esta sección se deberá introducir la ruta donde se desea guardar los archivos comprimidos y los descomprimidos, además del estilo de organización de las carpetas en dichas unidades, pudiendo escoger entre 3 opciones. Es importante la elección de una unidad donde se tengan permisos de lectura y escritura y una ruta de pocos caracteres. Cabe destacar que estas configuraciones también se pueden eliminar y actualizar, aunque una vez establecidas no es recomendable. Este apartado es de carácter obligatorio y sin estos datos no podrá dar inicio la descarga.

DOWNLOAD CONFIGURATION

En esta opción se podrán elegir los parámetros de descarga de cada misión Sentinel con imágenes disponibles siendo la decisión en la Figura 2. Al seleccionar cualquiera de estos se abrirá un formulario donde se deberán rellenar los parámetros de descarga que se indican en la pantalla como se representa en la Figura D3

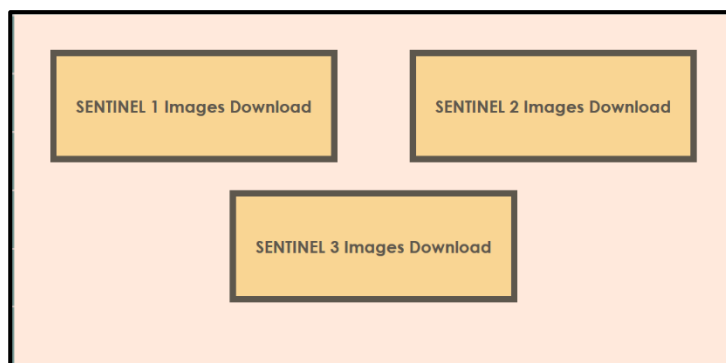


Figura D. 2: Tipos de descargas en Download Configuration

El formulario se divide en una barra superior con el logo de un satélite y el título "SENTINEL PRODUCT INGESTOR". A la izquierda hay un menú con las siguientes opciones: Path Configuration, Download Configuration, Study Tools, Status, Add Users y Exit. El contenido principal está titulado "Sentinel 2 download parameters" y contiene los siguientes campos:

- Launch Hour:
- Start Date:
- End Date:
- Minimum longitude:
- Maximum longitude:
- Minimum latitude:
- Maximum latitude:

En la parte inferior del formulario hay dos botones: "LAUNCH" y "RESET". En la barra inferior del formulario hay un botón "Return".

Figura D. 3: Parámetros a rellenar en Download Configuration

Una vez escogidos los parámetros, se volverá a la Figura D.2. Si a continuación se vuelve a seleccionar la descarga de la que se acaba de establecer la configuración, aparecerá en pantalla el estado de la descarga (run o stop) y el tipo de orden establecido (default o set), además de cuatro opciones: eliminar, parar, continuar y actualizar y establecer árbol.

Pulsando sobre *Set Preferences Download*, se abrirá un nuevo formulario de carácter opcional donde se podrá establecer el orden jerárquico de los repositorios disponibles para esa colección en concreto, además de los tiempos máximos permitidos de descarga.

Figura D. 4: Selección del orden de los repositorios web en Download Configuration

Las funciones de las restantes opciones se intuyen por sus nombres por lo que no procede su explicación.

STUDY TOOLS

Al escoger esta sección del menú aparecerá en pantalla los dos tipos de estudios a realizar. La función de esta opción es la de establecer los parámetros para el lanzamiento de los estudios de descarga y/o adquisición.

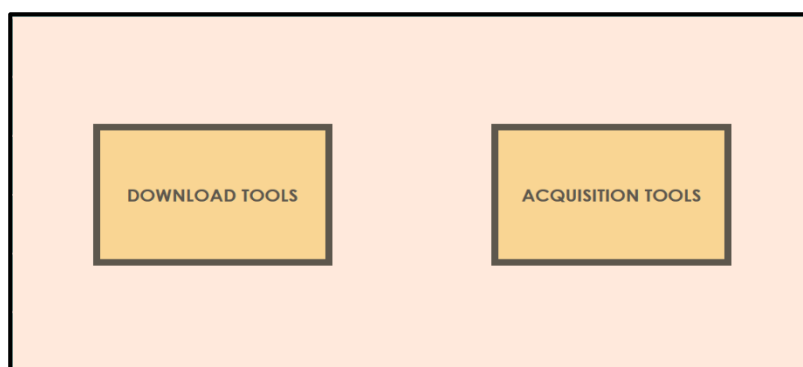


Figura D. 5: Tipos de estudios en Study Tools

El mecanismo al pulsar sobre una opción u otra se asemeja a la explicada en *DOWNLOAD CONFIGURATION*, pero sin la opción de escoger la hora de ejecución ni posibilidad de configurar una jerarquía ya que el estudio lanzado se pone a correr para todos los servidores.

STATUS

La ventana de estado servirá para la monitorización del sistema teniendo 4 opciones, aunque actualmente 3 operativas: Imágenes guardadas, Registro de errores y Registro de logs. Seleccionando sobre cualquiera de éstas, se abrirá una página con el listado correspondiente.

List of logs on the download process

2017-06-18 - 23:15:36: sentinel_3_download created, ready to start at 06:00:00

2017-06-18 - 23:20:36: sentinel_2_download created, ready to start at 04:00:00

2017-06-18 - 23:20:36: sentinel_3_download has been deleted, process finished

2017-06-18 - 23:25:36: sentinel_2_download has been deleted, process finished

2017-06-19 - 16:33:28: S2A_MSIL1C_20170227T095021_N0204_R079_T33UXP_20170227T095613 downloaded with no errors from PEPs

2017-06-19 - 16:35:13: S2A_MSIL1C_20170313T225521_N0204_R001_T02VNR_20170313T225523 downloaded with no errors from PEPs

2017-06-19 - 17:00:27: S2A_MSIL1C_20170227T095021_N0204_R079_T33UXP_20170227T095613 downloaded with no errors from PEPs

2017-06-19 - 17:01:58: S2A_MSIL1C_20170313T225521_N0204_R001_T02VNR_20170313T225523 downloaded with no errors from PEPs

2017-06-19 - 19:12:17: S1B_JW_GRDH_1SDV_20170618T174638_20170618T174703_006108_00ABA2_1ADD downloaded with no errors from SCIHUB

Search by date:

Return

Figura D. 6: Ejemplo del interfaz de seguimiento de logs

ADD USERS

Como el propio nombre indica, se trata de la opción a través de la cual el administrador puede añadir y ver los usuarios de esta aplicación.

Al finalizar, por seguridad se debe cerrar la sesión.

New User

Name:

Surname:

Username:

Password:

Role:

Users List

D424665 - Ainhoa Rojo

D011735 - Laura López

Search by name:

Figura D. 7: Pantalla de configuración de nuevos usuarios